TYPO3

Little Gems in TYPO3 v13

Features and changes that didn’t make the headlines

Jigal van Hemert
jigal.van.hemert@typo3.org

@jigal@fosstodon.org

@jigal

Who is this guy?

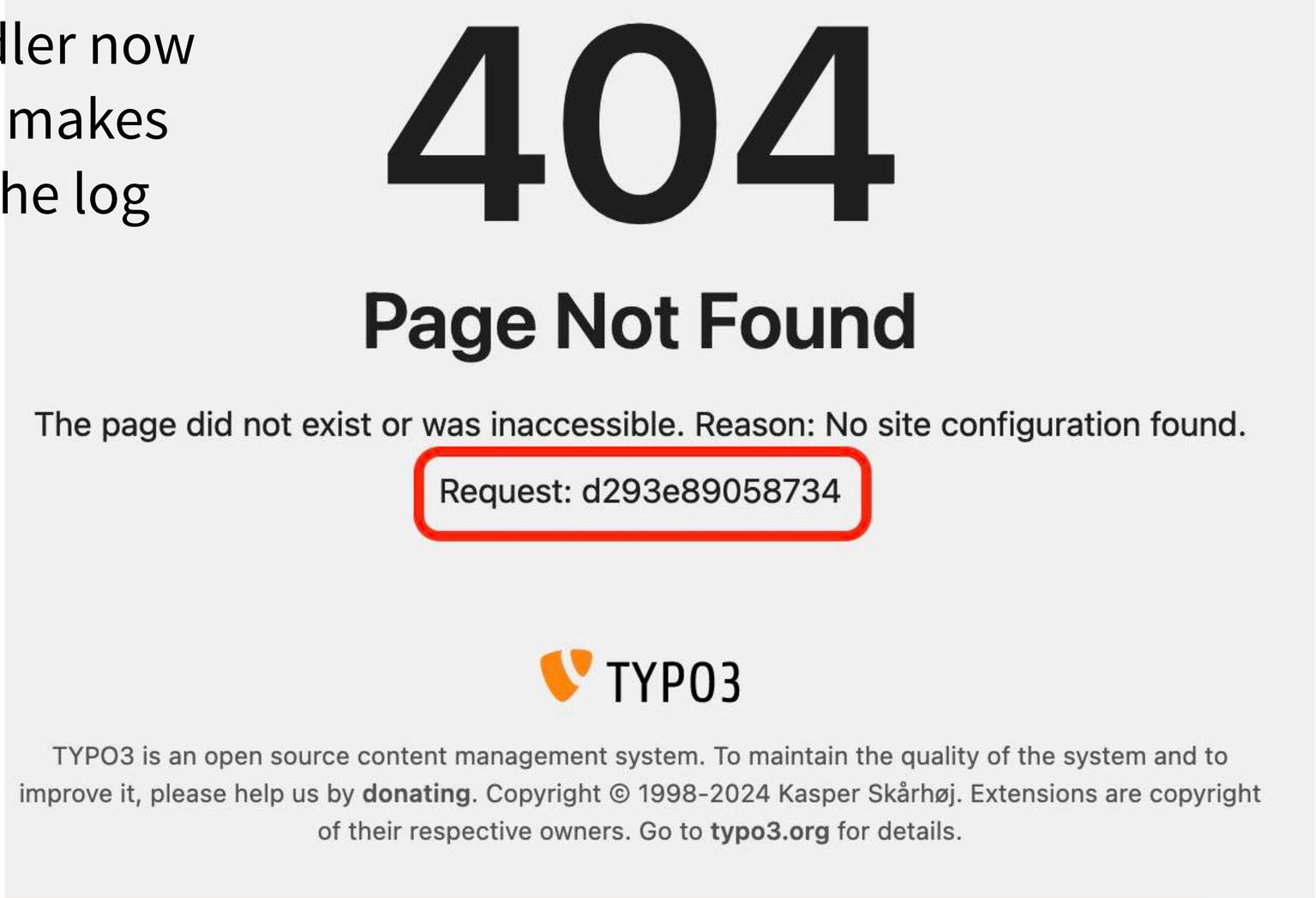
- TYPO3 developer at MaxServ
- Working with TYPO3 since v3.8 (?)
- Translations of core labels in Dutch
- Shares home with wife and ten dogs

maxserv
DIGITAL MASTERY

Frontend

Request ID in error message

The production exception handler now also shows the request ID. This makes it easier to lookup the error in the log and find the details.



SVG scaling / cropping

Cropping based op “crop” argument.

Result in processed file as SVG. Rasterising still possible by forcing image format.

Wrappers are created to apply the viewbox in the processed file.

Static route type “asset”

You can define a route to a public asset.

```
routes
  -
    route: directions.pdf
    type: asset
    asset: ‘EXT:my_ext/Resources/Public/Pdf/
directions.pdf
```

Routing variables can be declared ‘static’

Routing parts that would normally be dynamic can be configured as ‘static’ to avoid cHash URL parameters

```
routeEnhancers:
  Verification:
    type: Simple
    routePath: '/verify/{code}'
    static:
      code: true
    requirements:
      code: ‘[a-f0-9]{40}'
```

new RedirectLoginErrorHandler

A 403 error can now be handled with a redirect (e.g. by felogin)

TypoScript / TSConfig

Null coalescing operator for TypoScript constants

If you want to use new constant names but still need to support the old names there is a new operator `??` . It uses the first constant in the list that has a value.

```
plugin.tx_myext.settings.size = {$config.size ??
$myext.setting.size}
```

First `config.size` is checked and if that's not set the new `myext.setting.size` will be used. It supports multiple variables:

```
something = {$new.constant ?? $old.const ??
$really.old.const}
```

Application context in TS data

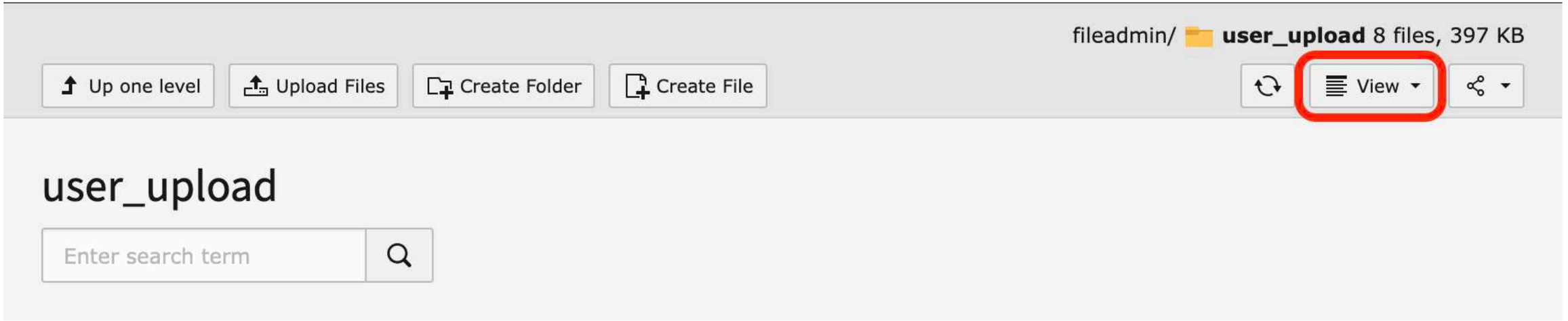
```
if {
   value.data = applicationcontext
   equals = Production
}
```

(It was already available in conditions)

User TSConfig default resources view mode

In resource overviews users can select between tiles and lists. Default can be set in User TSConfig

```
options.defaultResourcesViewMode = list
```



User TSConfig default search level in List module

The available search levels can already be configured

`mod.web_list.searchLevel.items`

The default can now be set with

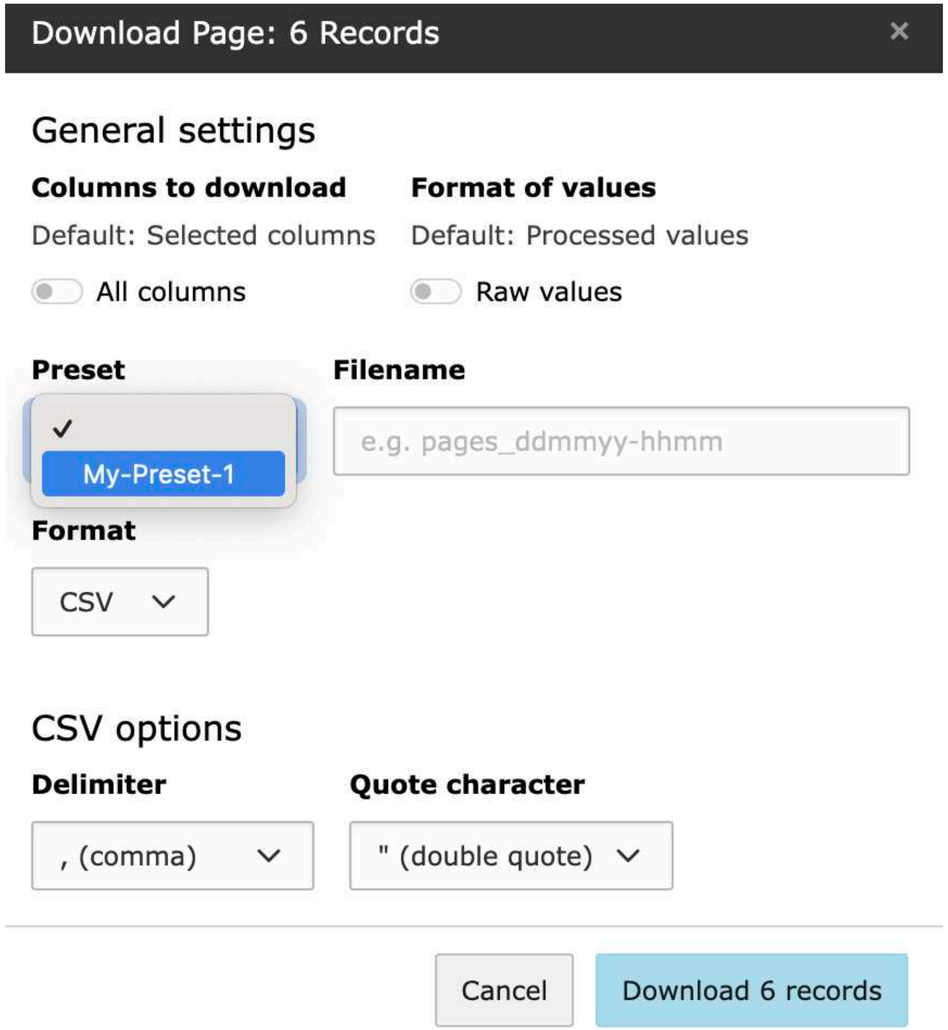
`mod.web_list.searchLevel.default`

Download presets in TSConfig (1)

The download popup in the List module can now have presets which are defined in TSConfig. Multiple preset per table are supported.

```
mod.web_list.downloadPresets {
  pages {
    10 {
      label = My-Preset-1
      columns = uid, title, crdate, slug
      identifier = my-preset
    }
  }
```

Download presets in TSConfig (2)



Cachebusting for EXT: resources

If a resource in an extension is changed browser may use the old version from cache.

```
10 = TEXT
10.value = { asset : EXT:my_ext/Resources/Public/Icons/
Extension.svg }
10.value.insertData = 1
```

parseFunc allowTags and denyTags are optional

The properties allowTags and denyTags can be omitted if you don’t need them in the configuration. The HTML Sanitizer will already take care of safe HTML output.

PAGEVIEW object (1)

A new PAGEVIEW object, extension of FLUIDTEMPLATE, to render pages

```
page = PAGE
page.10 = PAGEVIEW
page.10.paths.50 = EXT:site/Resources/Private/Templates
```

Template folders for “pages”, “layouts”, “partials” are resolved (case insensitive), name of template is resolved from BE layout.

PAGEVIEW object (2)

Available variables:

- settings: TypoScript settings
- site: Site object
- language: Site Language object
- page: Page Information object (this contains id, pageRecord, pageLayout

record-transformation data processor

Data processor for `FLUIDTEMPLATE` and `PAGEVIEW`

Creates Record object with the relevant fields for the “type” according to TCA.

page-content data processor (1)

Collects content of entire page. Stored per identifier of a content area.

```
page = PAGE
page.10 = PAGEVIEW
page.10.paths.10 = EXT:site/Resources/Private/
Templates/
page.10.dataProcessing.10 = page-content
page.10.dataProcessing.10.as = allContent
```

page-content data processor (2)

In Fluid accessed via the alias, area identifier, ‘records’ each content element can be accessed

```
<f:for each="{allContent.main.records}" as="record">
  <div>{record.header}</div>
</f:for>
```

Toggles in context menu for custom doktypes

Since v10 doktypes > 200 are allowed. Context menu toggles now work for these page types.

New TypoScript config.sendCacheHeadersForSharedCaches

Use

```
config.sendCacheHeadersForSharedCaches = auto
```

instead of

```
config.sendCacheHeaders
```

to send a cache header with cache lifetime for proxy / CDN

Command Line

Option to force recreation of all processed files

```
bin/typo3 cleanup:localprocessedfiles --all
```

will clean the `sys_file_processedfile` table completely, forcing a rebuild of all processed files.

Create backend user groups

```
./bin/typo3 setup:begroups:default --groups Both
```

Only support for the two default groups for now.

Reference index update is much faster

Records pointing to deleted=1 records are removed first and soft-deleted ones are ignored. This increases the speed of the task considerably.

Generate Fluid ViewHelper schema

```
vendor/bin/typo3 fluid:schema:generate
```

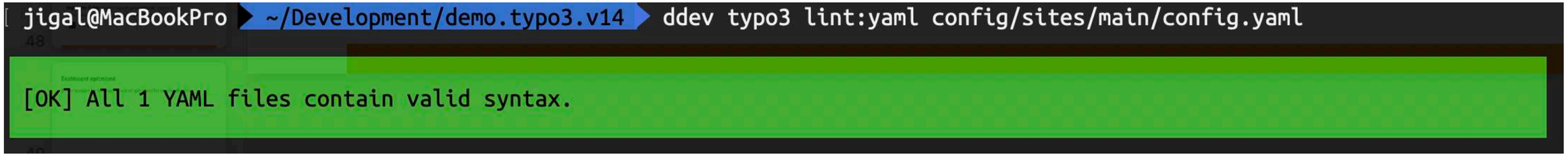
Collects information about all ViewHelpers in project and produces xsd files in `var/transient/`.

An IDE (e.g. PhpStorm) can use the information for autocomplete

Command lint:yaml

Existing functionality from symfony/yaml as command

`bin/typo3 lint:yaml`

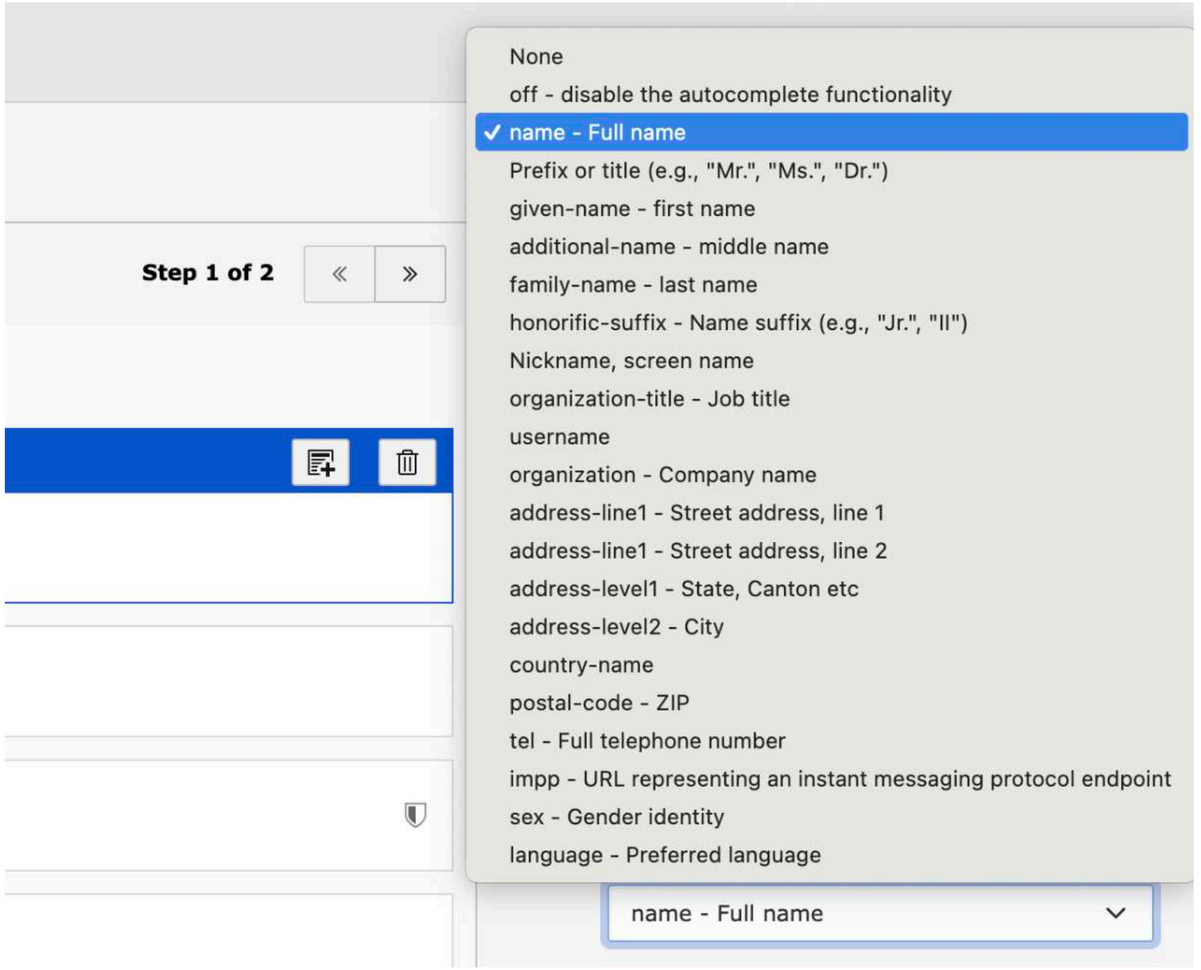


TYPO3
Backend

Autocomplete in forms can be set by editor

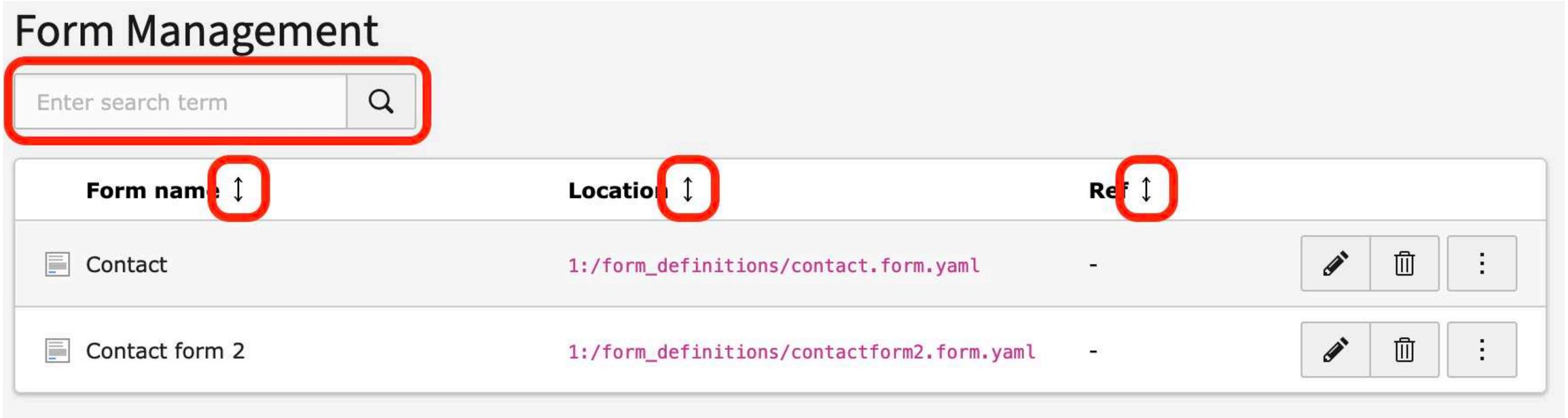
It was already possible to set the autocomplete attribute of form fields by editing the yaml configuration. This is of course not available to editors.

The GUI now allows editors to set relevant values for the autocomplete parameter.



Search and sorting option in Forms manager

You can now search for forms in the form manager. Also the list of forms can be sorted.



Reports module redirects check status improved

In the Reports module the entry about the redirects checks now shows if they were last checked more than 24 hours ago (configurable).

Redirects module filters and integrity status (1)

Redirects can be checked and the integrity status is stored.

Integrity status is shown in list of Redirects module.

Redirects module has filters for integrity status and ‘deleted’.

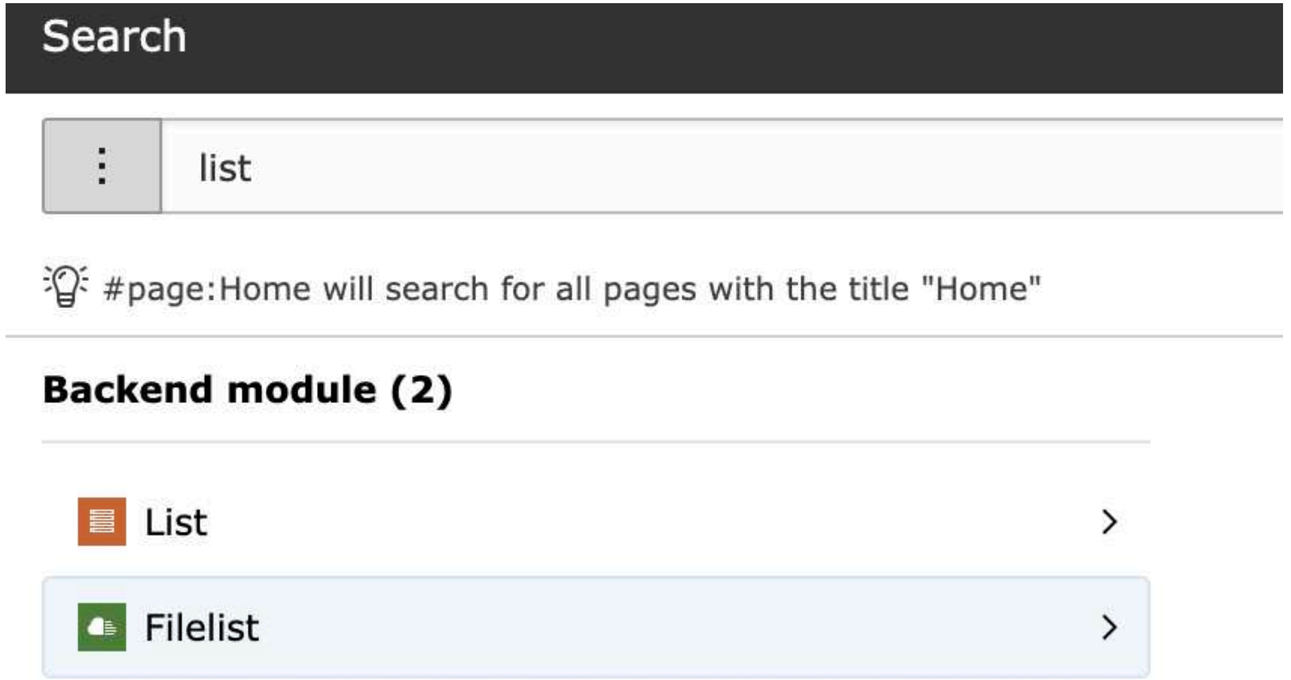
CLI command `redirects:cleanup` can now remove redirects based on status.

Redirects module filters and integrity status (2)

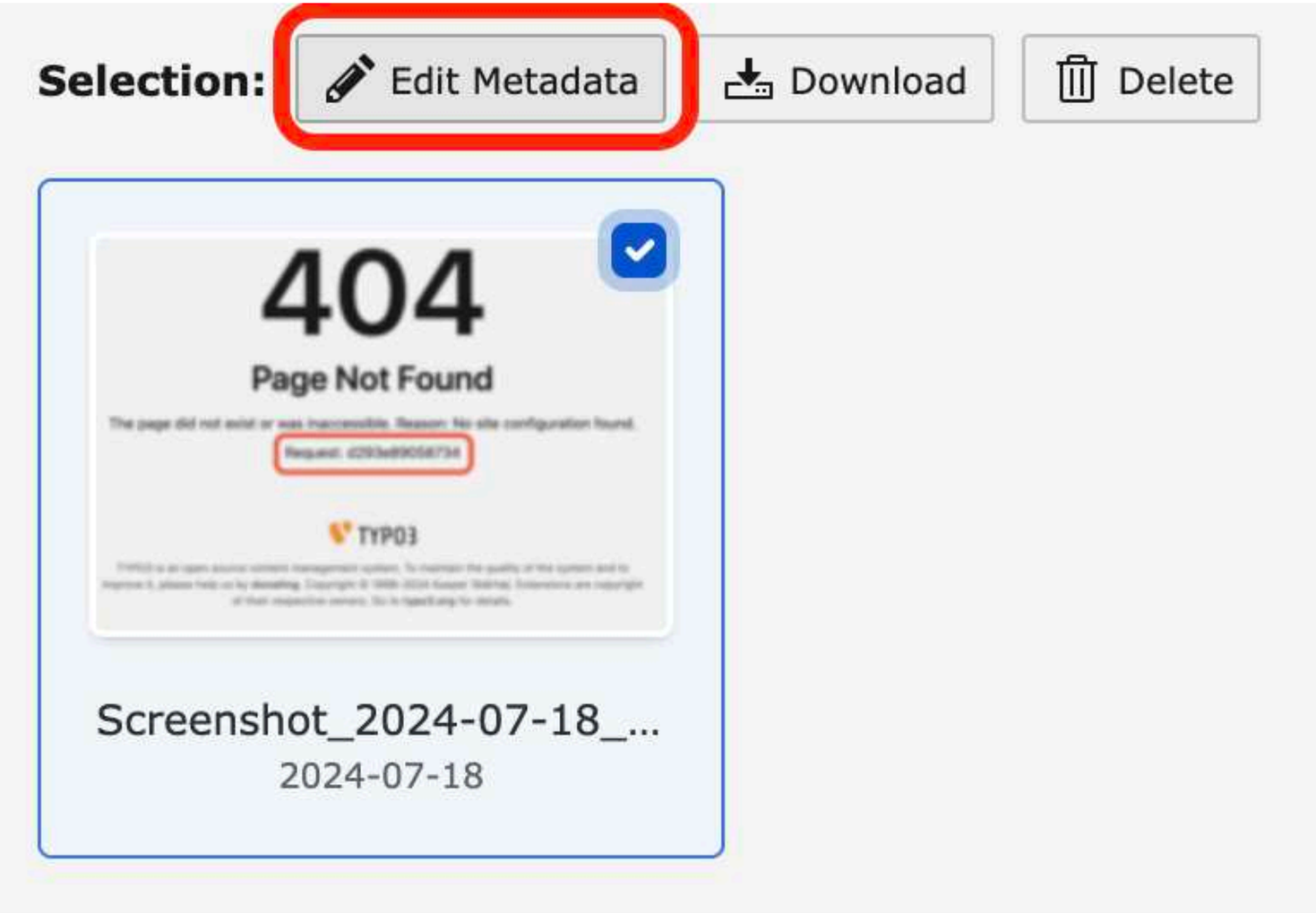
More items can be searched in LiveSearch

Modules to which you have access now appear in the results of LiveSearch.

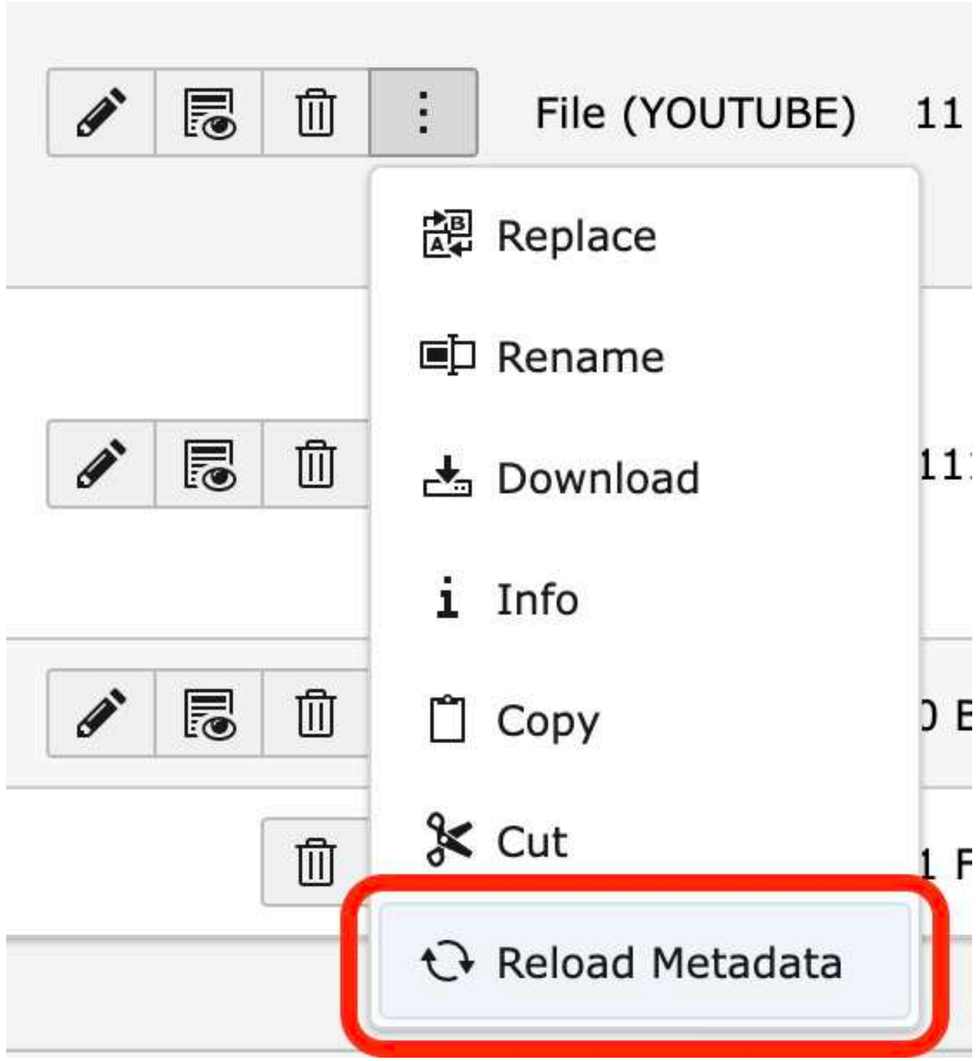
If Workspaces is installed you can also find workspaces you have access to.



Filelist: Edit metadata button in element information view



Action ‘Reload Metadata’ for online media

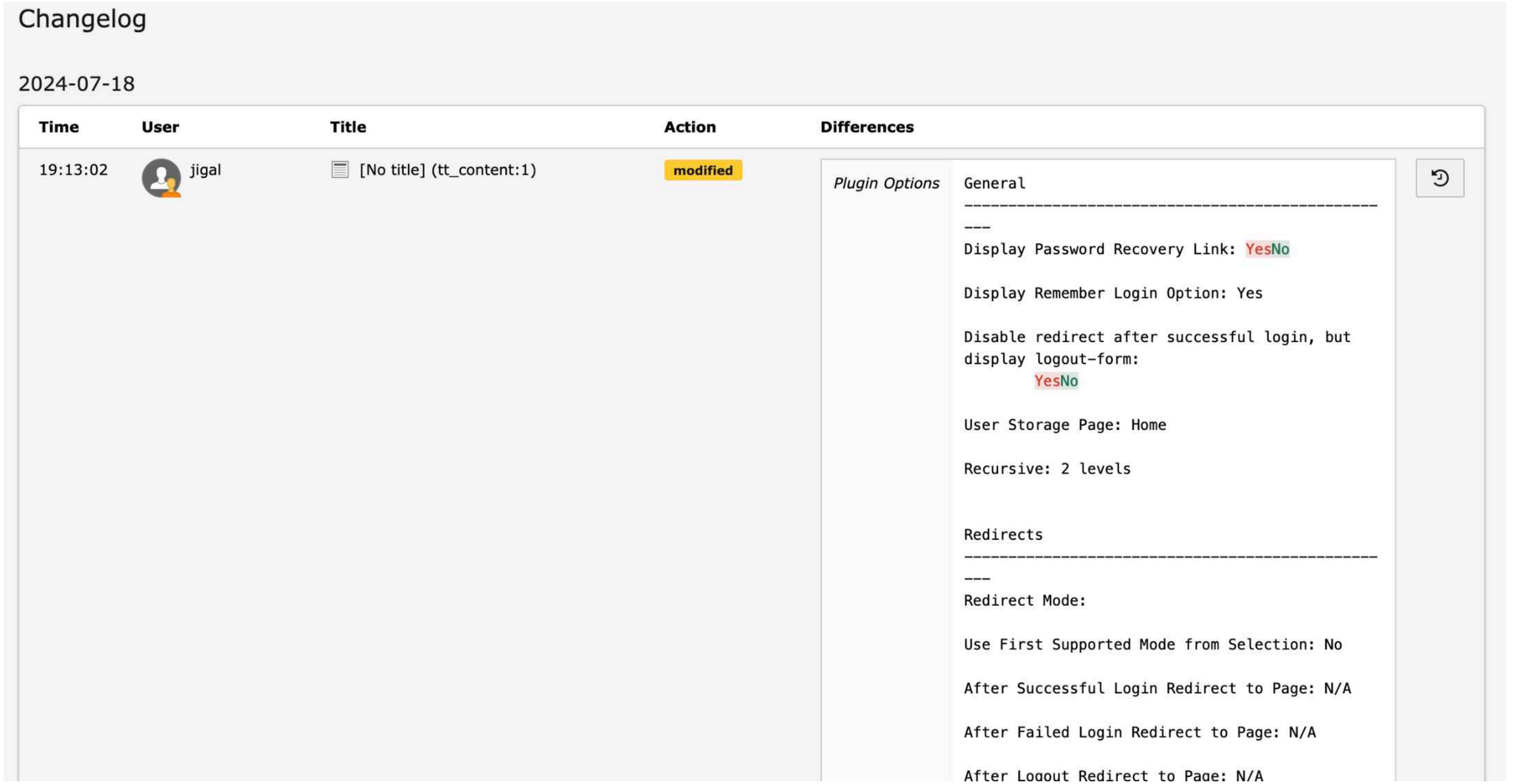


Flexform values are shown human readable (1)

In history and workspaces the diff of a flexform field is shown as key-value pairs. This makes it far easier to see the differences.

The flexform is parsed according to the datastructure to display the values instead of the raw XML.

Flexform values are shown human readable (2)



Hotkey API

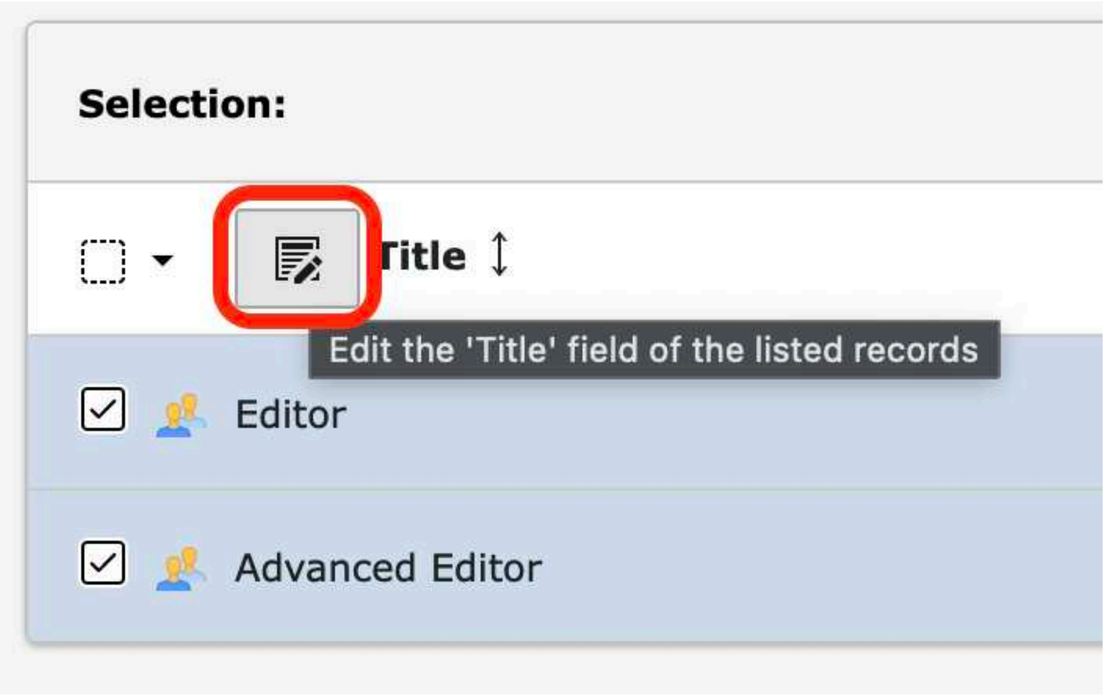
In JS hotkeys can be defined

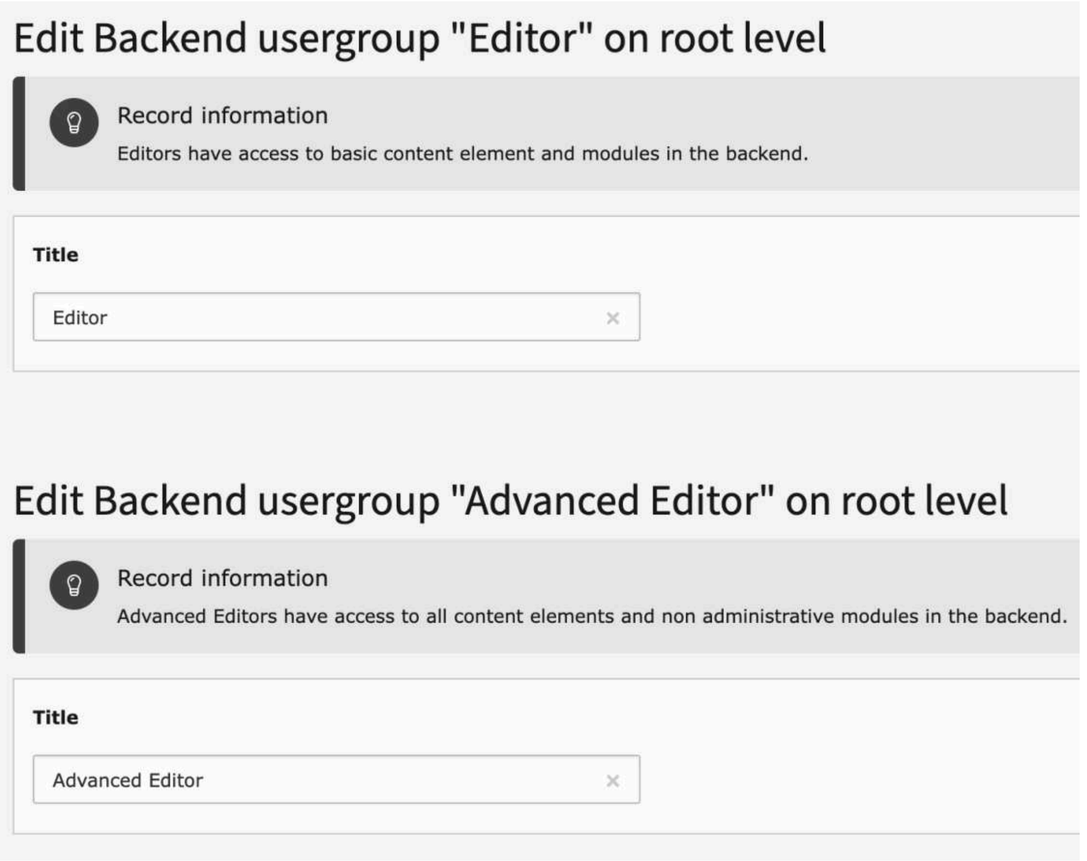
Possible to restrict to a scope (a module). ‘all’ scope for global use.

ctrl + K (⌘ + K) : open LiveSearch
ctrl + S (⌘ + S) : save document in FormEngine and Scheduler module
shift + ctrl + S (⇧ + ⌘ + S): save & close document in FormEngine and Scheduler module

List module: Edit specific columns for multiple records

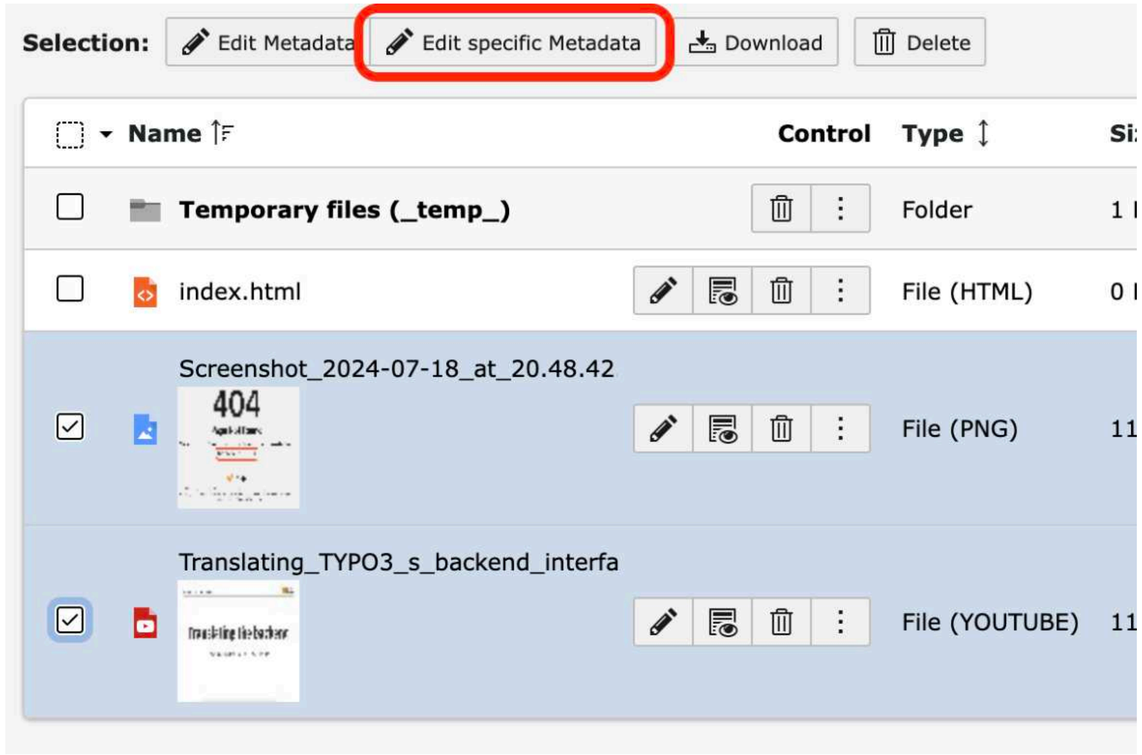
In the List module columns can be selected for the list. When some records are selected it’s now possible to only edit those columns for these records at the same time.

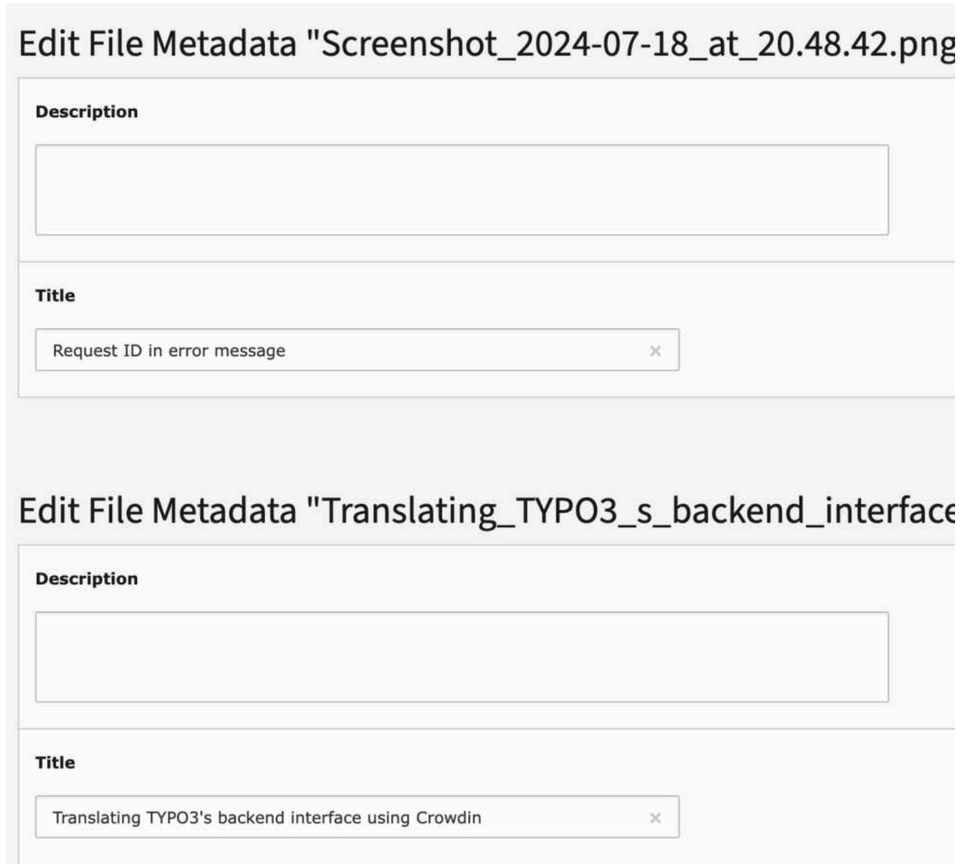




Filelist module: Edit specific columns for multiple records

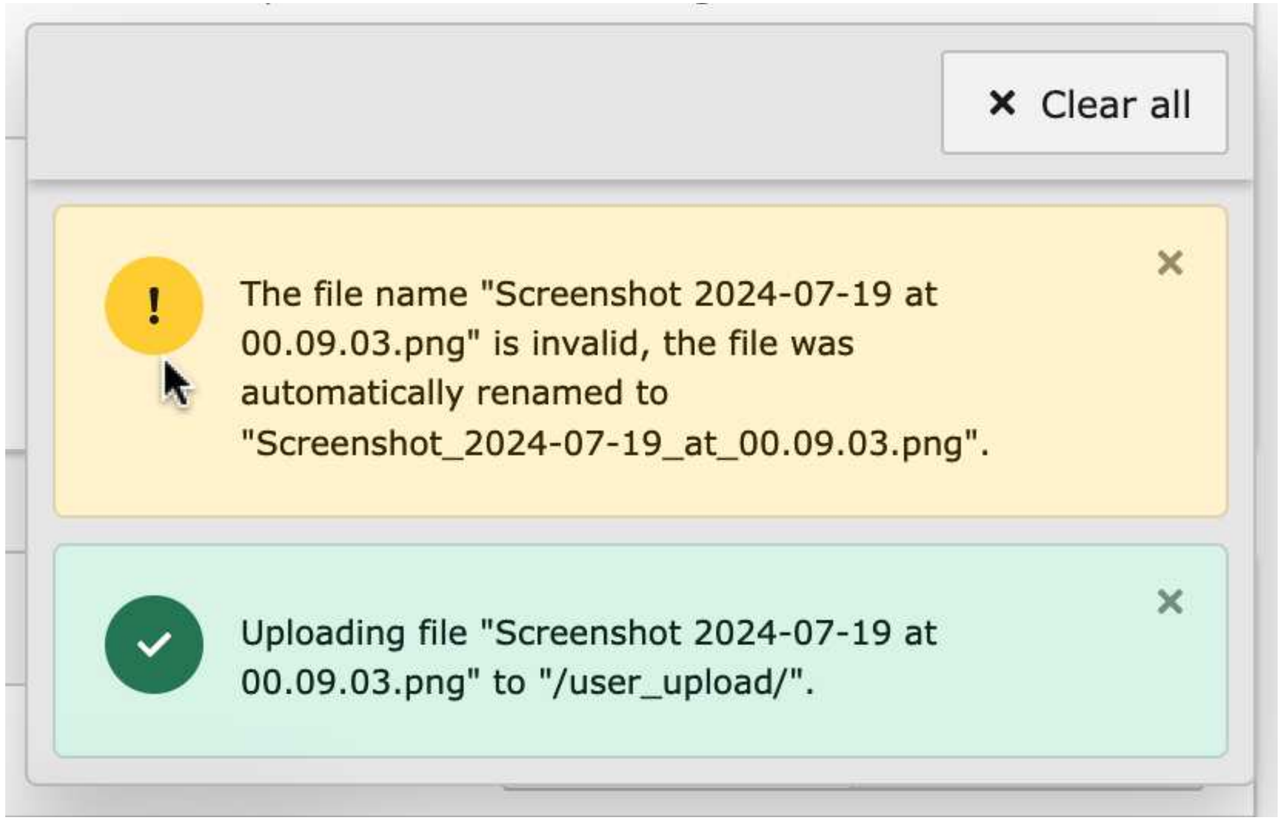
In the Filelist module metadata columns can be selected for the list. When some files are selected it’s now possible to only edit those metadata columns for these files at the same time.





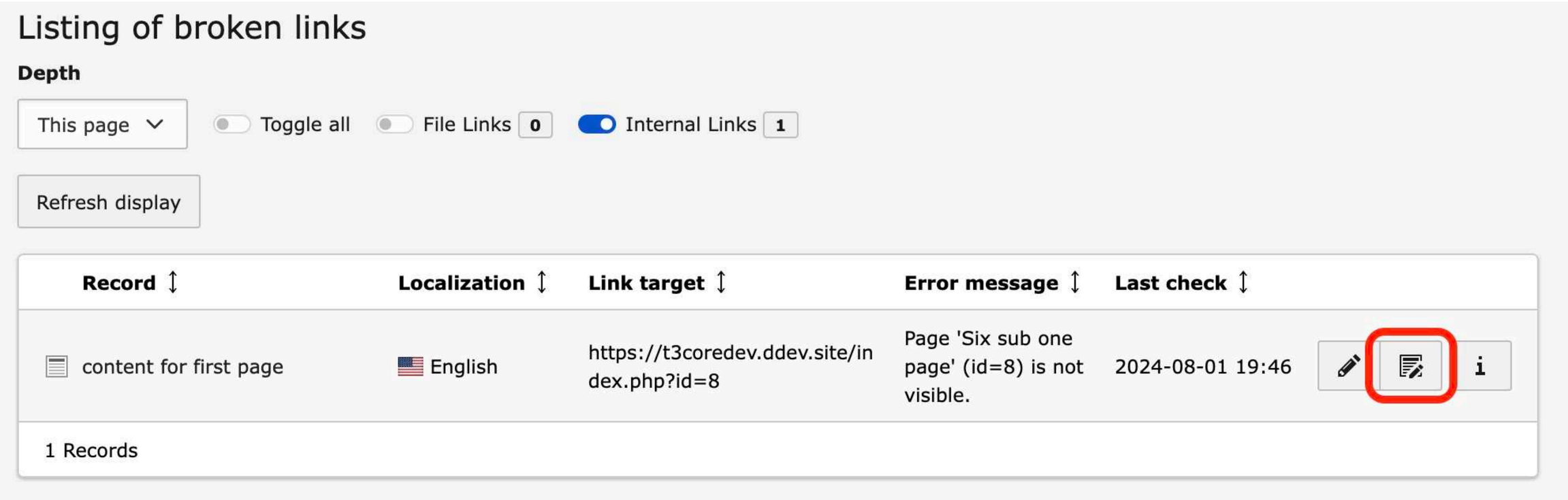
Notification container improved

The notification container in the bottom right corner is scrollable. If there are multiple notifications then a **Clear all** button is also available.

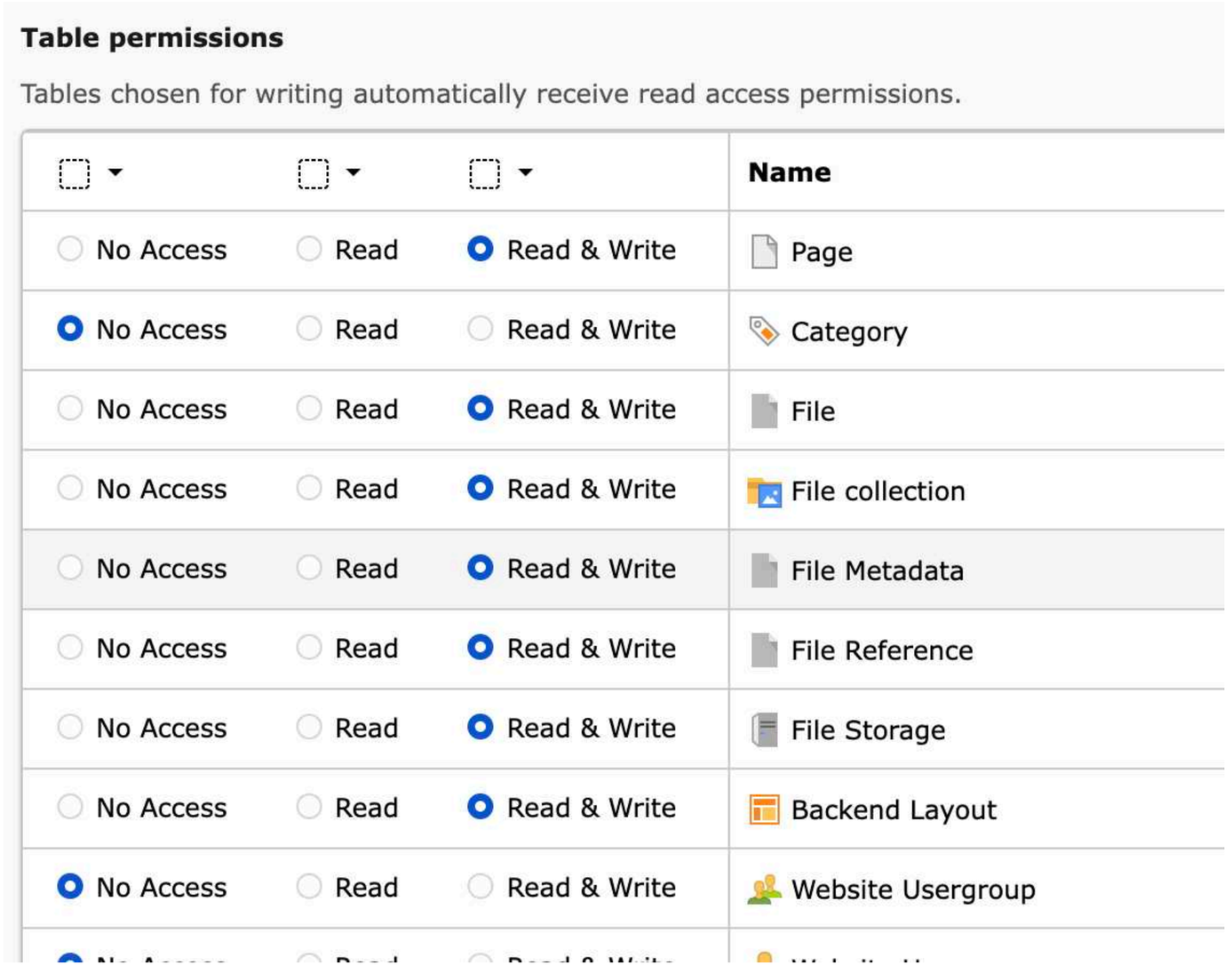


Edit record in LinkValidator

Besides the button to edit the field where the invalid link is located there is now an extra button to edit the entire record. This provides more context.



Read/write table access in BE groups editor combined



Backend URL configurable

Domain/URL: `$GLOBALS['TYPO3_CONF_VARS']['BE']['entryPoint']`

Legacy entry point can be disabled in `composer.json`

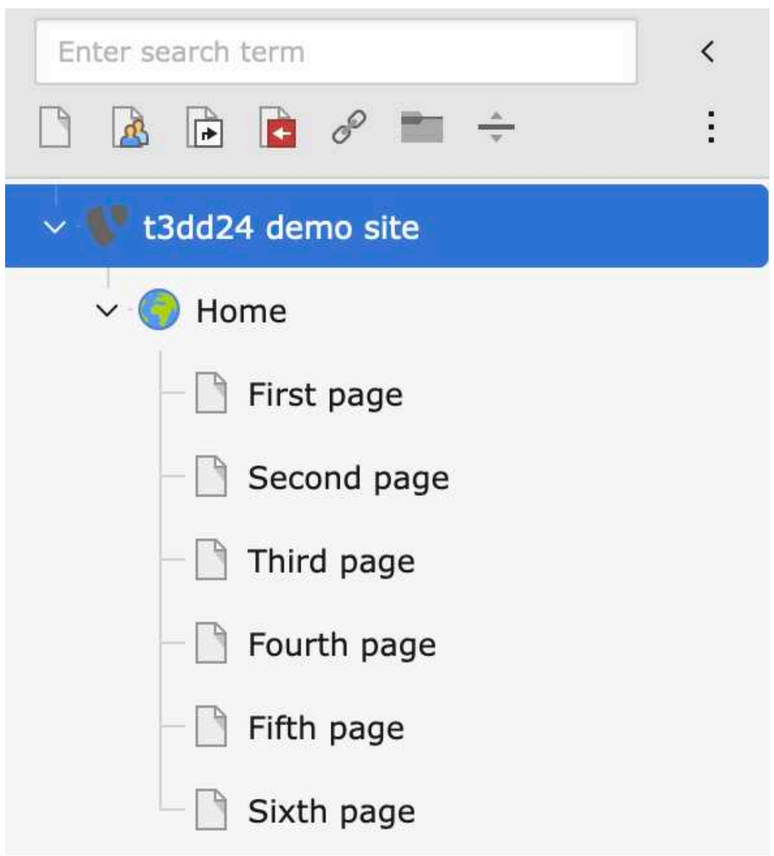
It's only routing, so Install Tool and assets still use `/typo3/`

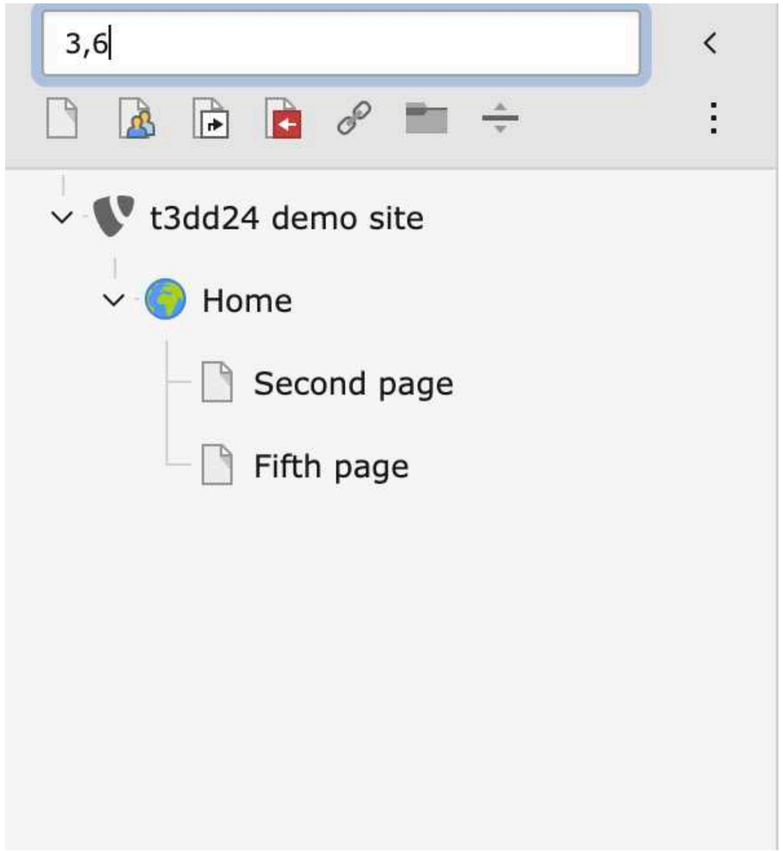
Page tree (and others) are now HTML/CSS (again)

Previously the trees were SVG based. To support RTL text and dark mode the trees are now HTML/CSS based (again).

Page tree search supports multiple UIDs

A comma separated list of UIDs can be used to search for multiple UID values at the same time.



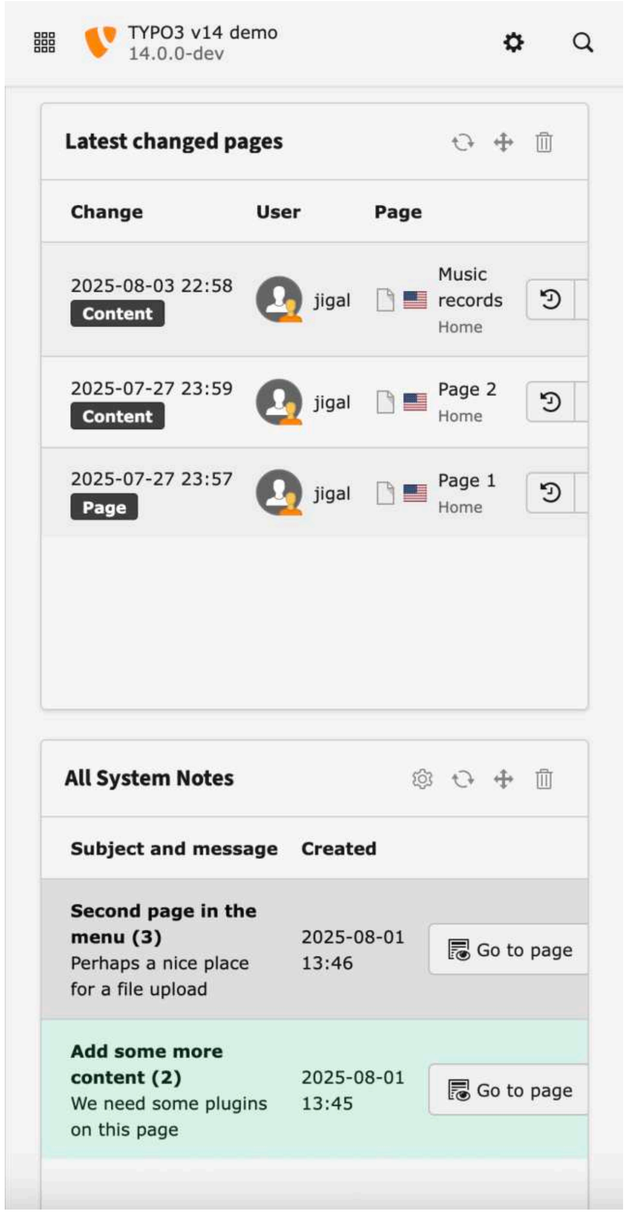


new properties in BE layouts

- identifier: readable alias of colPos
- slideMode: slide / collect / collectReverse

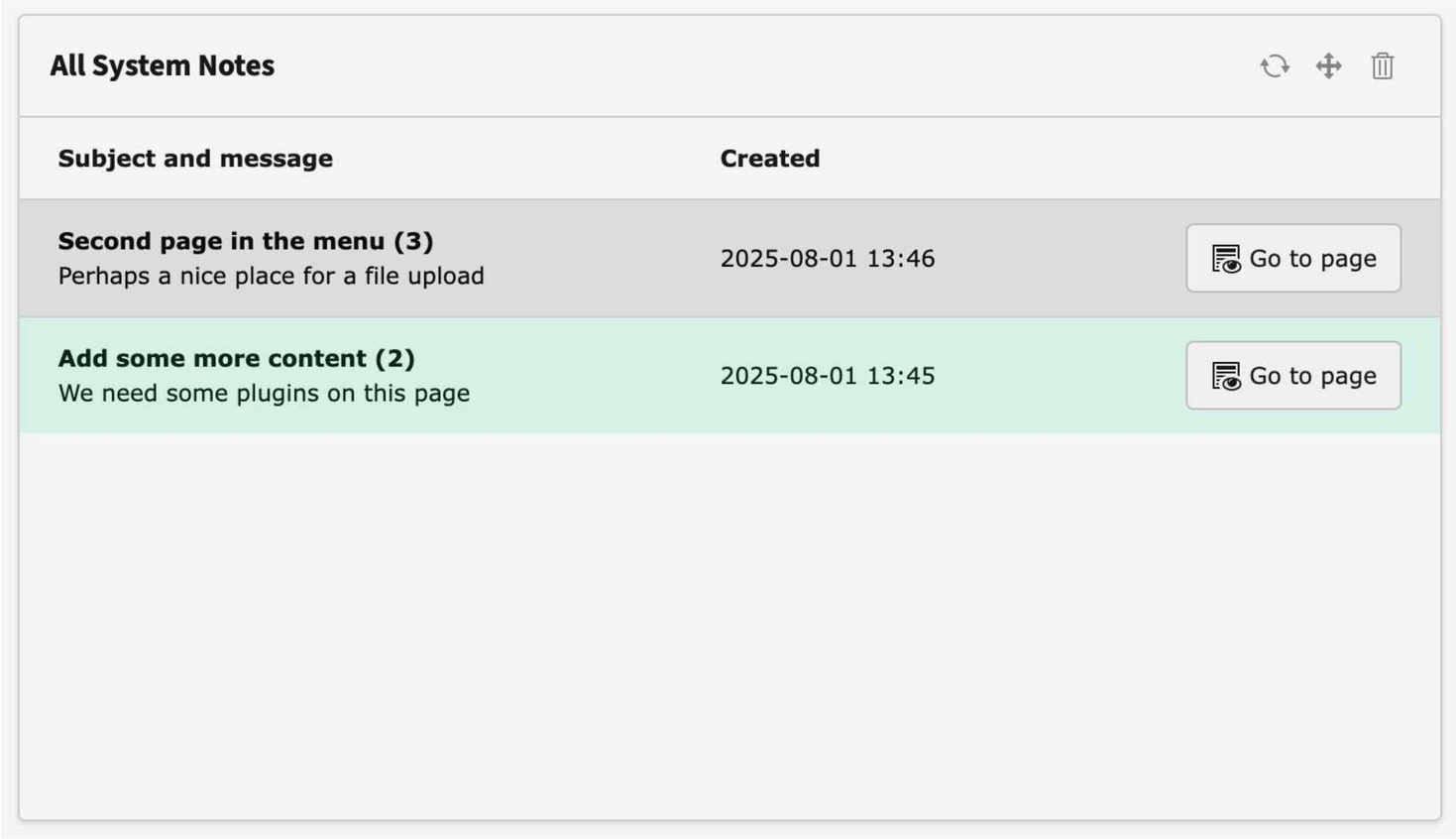
Dashboard optimized

Better support for keyboard use and optimized for smaller devices.



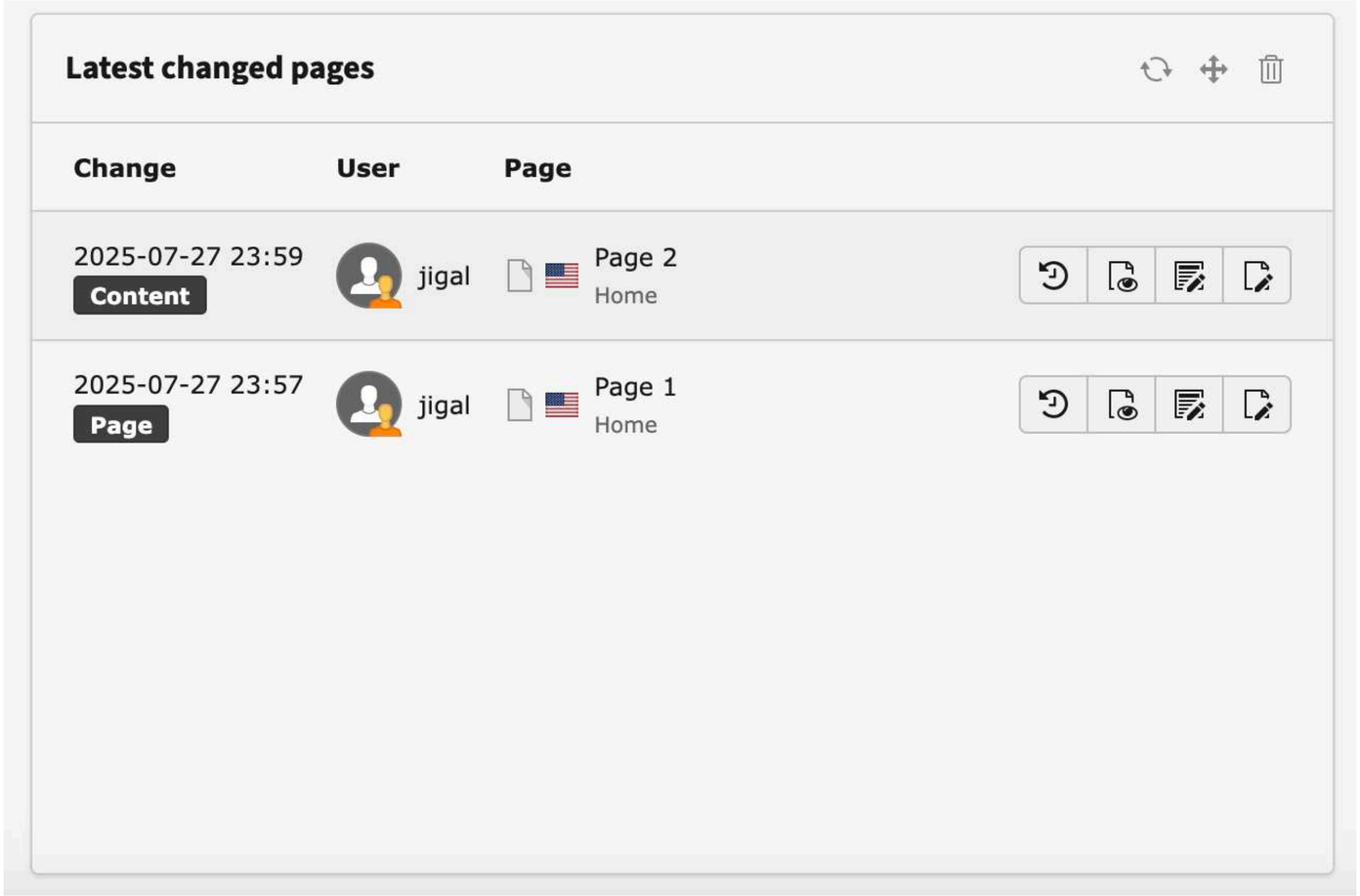
Dashboard widget to list internal notes

BE user with access to internal notes table can use widget to display internal notes on the dashboard.



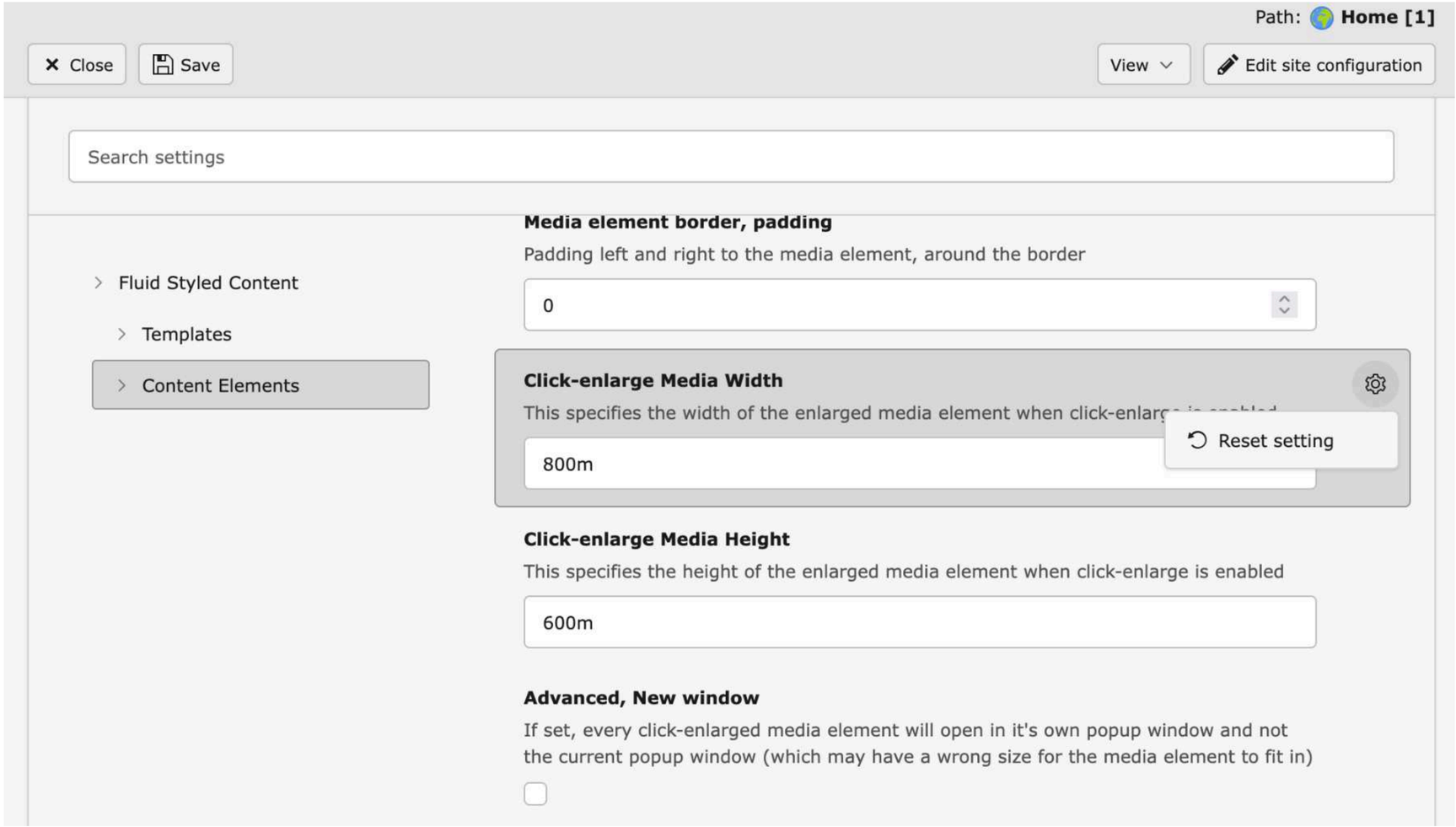
Dashboard widget to list latest changed pages

Another new widget, now one with recently changed pages



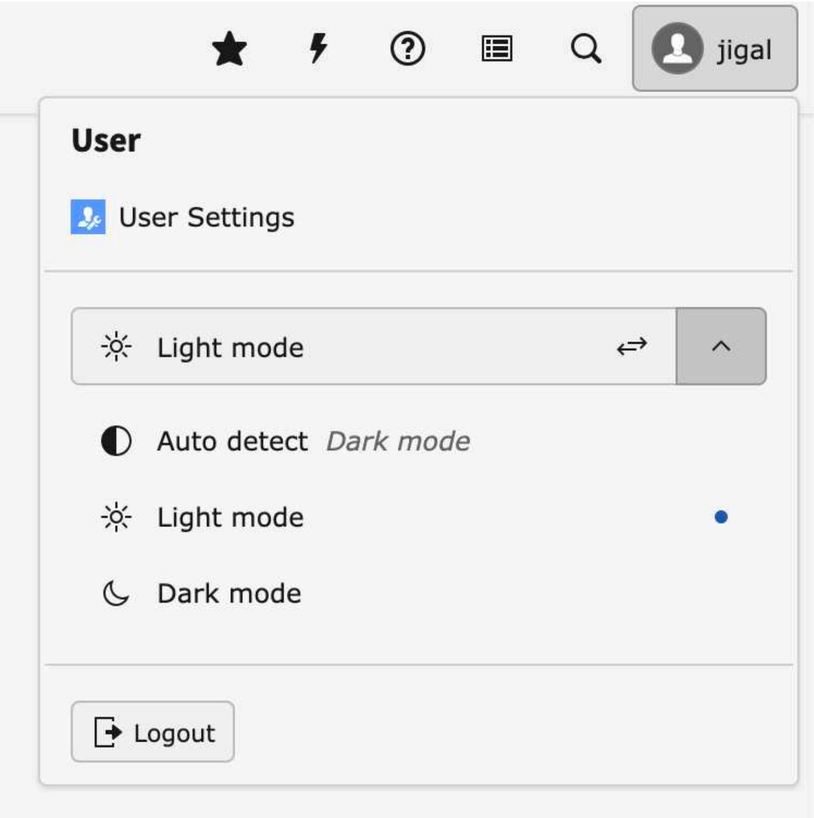
Site settings editor

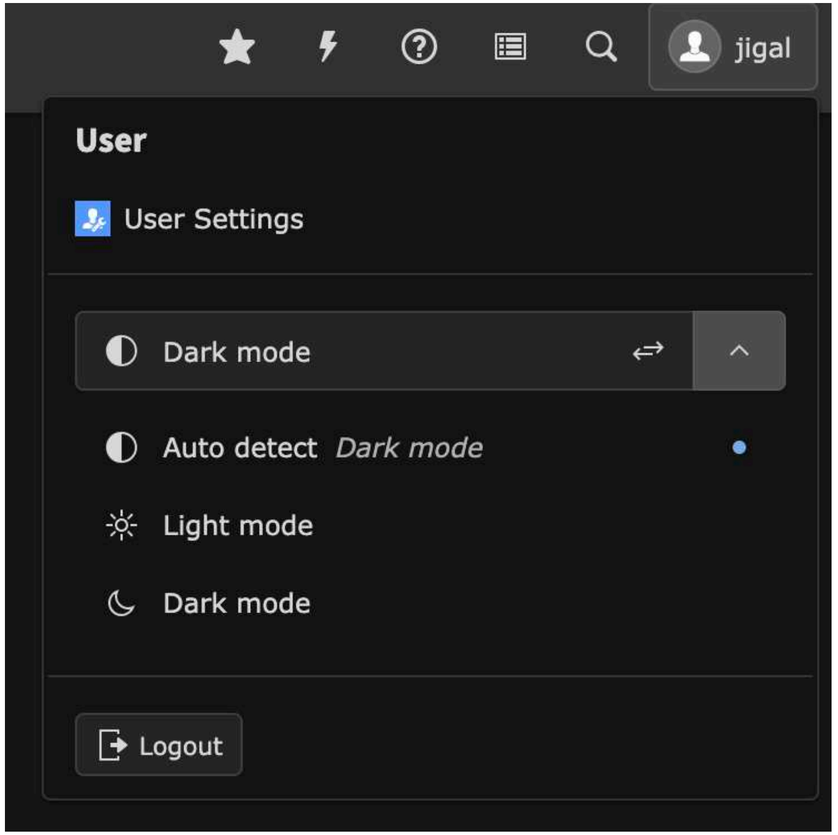
New editor to edit settings for a site with support for site sets. It only stores the values that are different from the defaults.



Switch between dark and light mode

Both in the user settings and in the user dropdown in the backend interface you can switch between light and dark mode.





Link validator checks more fields

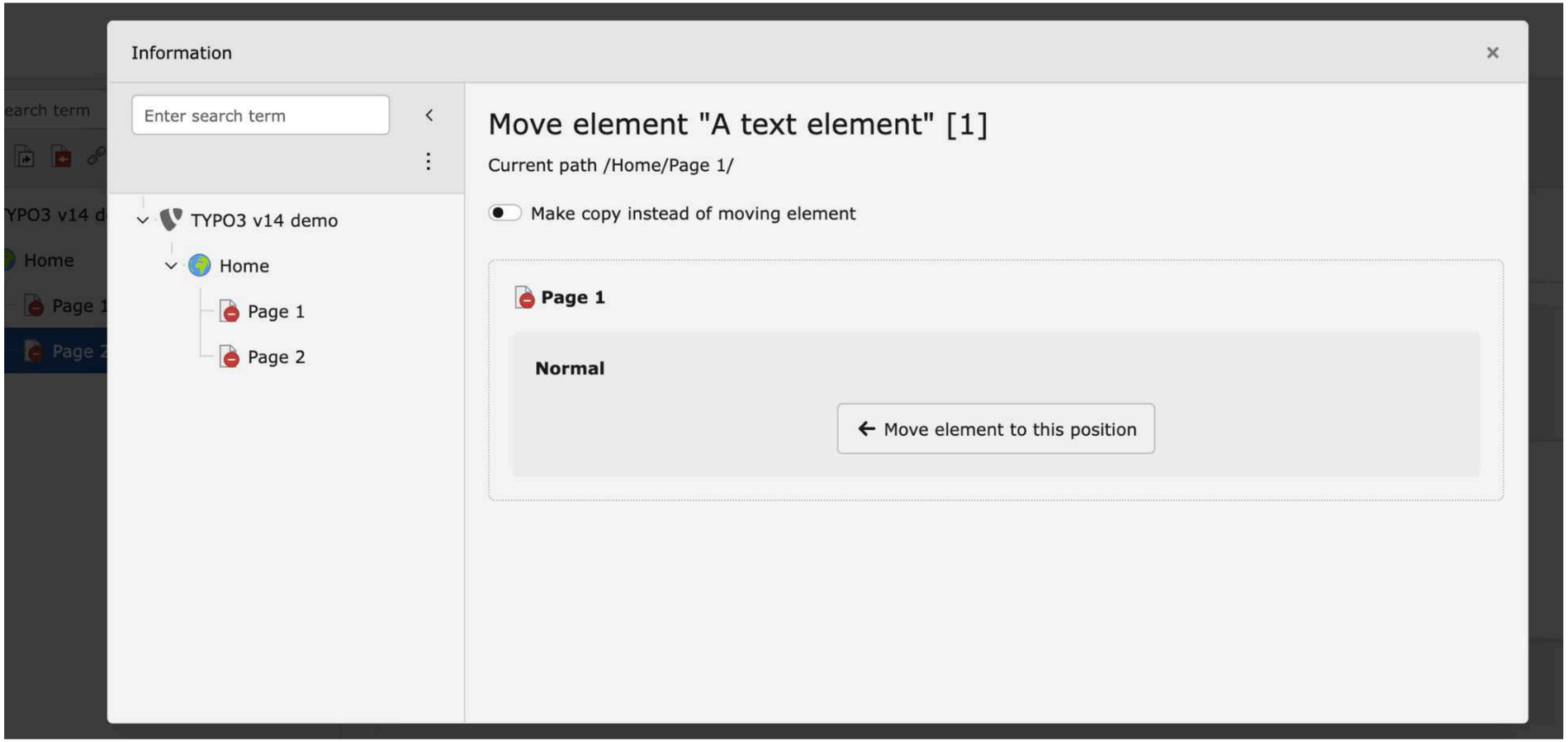
The fields

- pages.canonical_link
- sys_redirect.target
- sys_file_reference.link

are now checked by default by the link validator

Move content to page by dropping it on pagetree

Move element to page in pagetree. Modal opens for position on page.



Color preview in various places

In FormEngine™, TypoScript constants, Site settings, extension settings the chosen color is displayed.

TYPO3

Miscellaneous

Disabled PHP functions configurable

In the Install Tool it’s possible to add PHP functions to the check for disabled functions.

The check results in a warning if it fails, not in an error anymore

```
$GLOBALS[‘TYPO3_CONF_VARS']['SYS']
['allowedPhpDisableFunctions']
```

CTE to speed up rootline fetching

A **Common Table Expression** (CTE) is used to fetch a rootline in a single query.

First version supports detection of cyclic recursion and workspaces. In the future language handling and mount points could be included in the CTE.

A CTE is like a named subquery which can be used like a table in the main query. Multiple CTEs can be used in a query.

CTE can be used in another CTE (“nested”) or can be made recursive.

Extra “internal” and “experimental” functions in queryBuilder

New languages

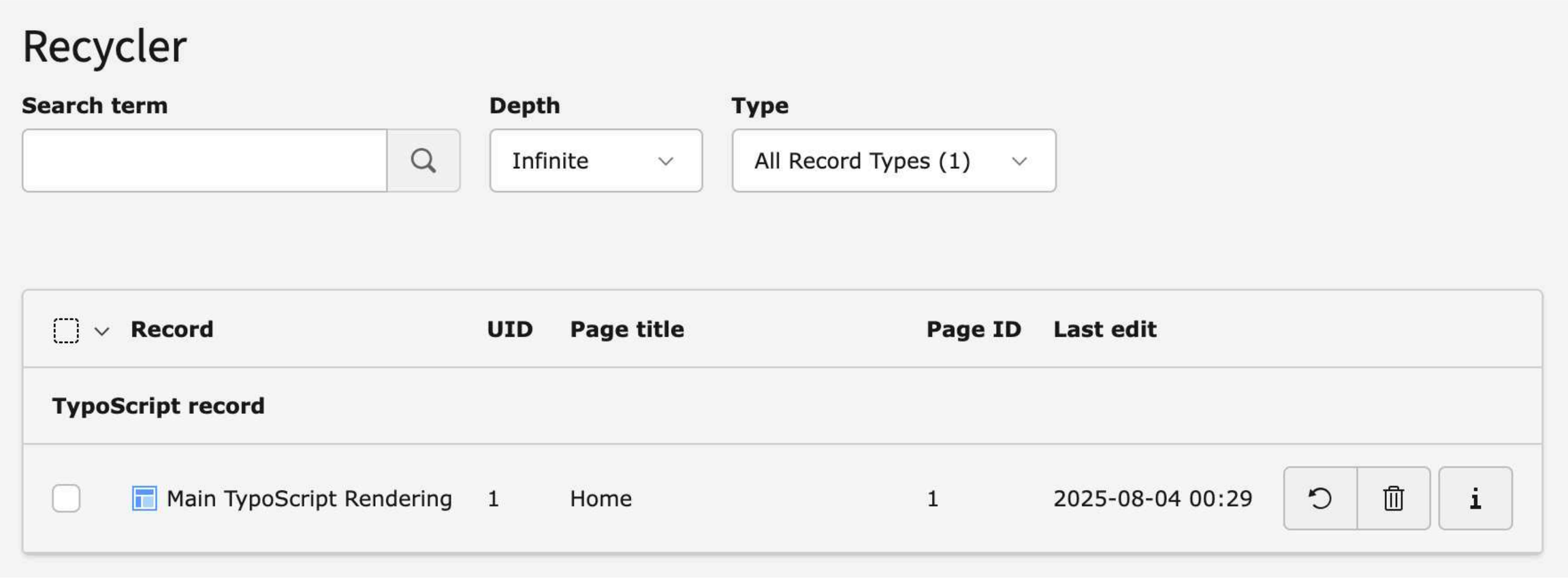
- Maltese (mt)
- Gaelic (gd) (Scottish Gaelic)
- Gaelic (ga) (Irish Gaelic)

Recycler page (doktype=255) removed

Recycler module leaves “deleted” items in their pid, recycler page was used by moving items to that page.

Recycler activated by default

For new installations



Fluid

Link/uri viewhelpers without Extbase

Just like the `f:link.page` (and `f:uri.page`) viewhelpers the `f:link.action` (and `f:uri.action`) viewhelpers can be used outside Extbase context.

In these cases all Extbase related parameters are required.

Scoped variable provider in Fluid

```
$varProvider = new ScopedVariableProvider(
  $renderingContext->getVariableProvider(),
  new StandardVariableProvider([‘myvar’=> $value])
);
$renderingContext->setVariableProvider($varProvider);
$content = (string)$renderChildrenClosure();
$renderingContext->setVariableProvider(
  $varProvider->getGlobalVariableProvider()
);
```

Cache busting EXT resources

If resources in extensions are replaced browser may use the old, cached version.

```
<f:uri.resource path=“EXT:my_ext/Resources/Public/Icons/Extension.svg" useCacheBusting="true" />
```

Attribute `useCacheBusting` is enabled by default and adds the “cache buster” as a query string or as part of the filename (depending on `FE/versionNumberInFilename`)

ViewHelper “feature”

ViewHelper checks if a feature switch is enabled

```
<f:feature name="some.feature">
  <f:then>Some Feature enabled</f:then>
  <f:else>Some Feature undefined or disabled</f:else>
</f:feature>
```

ViewHelper “split”

Split a string by a separator

```
<f:split value=“1,2,3” separator=“,”>
```

ViewHelper “join”

Joins an array

```
<f:join value=“{0:1,1:2,2:3}” separator=“, ”
separatorLast=“ and “>
```

Output:

```
1, 2 and 3
```

ViewHelper “replace”

Replaces one or more strings

```
<f:replace value=“Big feature” search=“Big”
replace=“Small“>

<f:replace value=“Big feature” search=“{0: ‘Big’, 1:
‘feature’}” replace=“{0: ‘Little’, 1:’gem’}”>

<f:replace value=“Big feature” replace=“{‘Big’:
‘Little’, ‘feature’: ‘gem’}”>
```

ViewHelpers “first” and “last”

Get first/last item from array

```
<f:first value=“{0:1, 1:2, 2:3}”>
```

Output:

```
1
```

ViewHelper “constant”

Returns PHP constant

```
<f:constant value=“PHP_INT_MAX”>

<f:constant
value=“\MyVendor\MyPackage\MyClass::CONSTANT_NAME”>
```

Arbitrary attributes for tag ViewHelpers (1)

All tag based ViewHelpers can use arbitrary attributes which are added to the resulting tag.

```
<f:form.textfield inputmode=“email”>
```

No need to register arguments that are simply passed to the outputted tag.

Arbitrary attributes for tag ViewHelpers (2)

No need to call `registerUniversalTagAttributes()` anymore.

If you do call `registerUniversalTagAttributes()` these attributes are available in `->additionalArguments[]` instead of `->arguments[]`

True, false and null as language elements

```
{true} , {false} , {null}
```

```
<f:render partial=“myPartial” optional=“{true}” />
```

More important for tagbased viewhelpers with properties like:

```
<f:form.textfield property="foo" disabled="0" />
```

```
<f:form.textfield property="foo" disabled="{false}" />
```

Turn image into base64 encoded data

The `<f:image>` and `<f:uri.image>` viewhelpers have attribute **`base64="{true}"`**

Adds image content as base64 encoded data to image tag

Embed file contents with asset viewhelpers

The `<f:asset.css>` and `<f:asset.script>` viewhelpers have a new option **`inline="{true}"`** that loads the file and embeds it in the page.

Fluid comment tags are hardened

The `<f:comment>` tags (and their contents) are removed before the template is parsed. No problems with syntax errors inside comments.

Development

Simple rotating file writer for logging

`RotatingFileWriter`

- set interval (default: daily)
- set max number of files (default: 5)

DI cache flushed in development context

When in development context “**Clear All Caches**” now also flushes the DI cache. This makes it easier to work with dependency injections while developing code.

PHP enums are supported by Extbase

PHP 8.1 introduced native enum support. Extbase now supports enums as properties of entities.

PHP attribute to autoconfigure event listeners (1)

- no need to register in Services.yaml
- can be used multiple times to register one listener class for several events
- can be used on class level and on method level (then this method is used instead of the default `__invoke()`)

PHP attribute to autoconfigure event listeners (2)

```
<?php
use TYPO3\CMS\Core\Attribute\AsEventListener;
#[AsEventListener(identifier: 'my-extension/null-
mailer-initialization', event:
AfterMailerInitializationEvent::class)]
#[AsEventListener(identifier: 'my-extension/null-
mailer-sent-message', event:
BeforeMailerSentMessageEvent::class)]

final class NullMailer
{
```

Controller attribute now AsController

```
use TYPO3\CMS\Backend\Attribute\AsController;

#[AsController]
class MyAwesomeController
{
```

Forward compatibility in v12

Auto configure message handlers with attribute

```
use
Symfony\Component\Messenger\Attribute\AsMessageHandler;

#[AsMessageHandler]
class MyAwesomeMessageHandler
{
```

DB field autocreated for datetime now bigint

Default is now bigint to support more dates.

Except for a few maintenance fields which can remain signed integer.

DB default values for TEXT, BLOB and JSON

Normal default values are not supported by MySQL for TEXT, BLOB and JSON fields. With MySQL 8.0.13+ default expressions are supported.

DBAL layer converts default values into expression syntax and vice versa

User TSconfig now loaded from file

Earlier the file `Configuration/page.tsconfig` could be used to load Page TSconfig.

Now the file `Configuration/user.tsconfig` is also supported.

The methods `EMU::addPageTSConfig()` and `EMU::addUserTSConfig()` are deprecated.

FE cache information request attribute

```
$GLOBALS[‘TSFE’]->no_cache
```

now

```
$request->getAttribute(‘frontend.cache.instruction')-
>isCachingAllowed();
```

Middleware can set it (and provide a ‘reason’; adminpanel will list reasons).
Exact way depends on position of middleware in stack.

FE page information request attribute

```
$pageInfo = $request
  ->getAttribute('frontend.page.information');
// Formerly $tsfe->id
$id = $pageInfo->getId();

// Formerly $tsfe->page
$page = $pageInfo->getPageRecord();

// Formerly $tsfe->rootLine
$rootLine = $pageInfo->getRootLine();
```

Item groups for foreign table

The TCA property ‘`foreign_table_item_group`’ lets you set a column name which contains an ID for a group of items.

In the property ‘`itemGroups`’ it’s possible to define labels for each item group ID.

Automatic registration content elements

All CType items are automatically registered as content element for both the New Content Element Wizard and the content element type select field.

The items in TCA config > items have fields for title, description, value, icon and group for use in the select box.

TSConfig is still supported, also for things like removing items from the wizard.

New querybuilder expressions (1)

- as(expression[, alias])
- concat(…parts)
- castVarchar(value, [length[, alias]])
- castInt(value[, alias])
- castText(value[, alias])
- repeat(numberOfRepeats, value[, alias])
- space(numberOfSpaces[, alias])
- left(length, value[, alias])
- right(length, value[, alias])
- leftPad(value, length, padding[, alias]

New querybuilder expressions (2)

- if(expression, truePart, falsePart [, alias])

Site sets

Contain settings, TypoScript, TSConfig and content blocks for a site

`Configuration/Sets/<setname>/config.yaml`

Sites can act as root template with TS from a set (`constants.typoscript` and `setup.typoscript` next to `config.yaml` are loaded)

`page.tsconfig` next to `config.yaml` is loaded

PHP enums used in several places

- icon sizes
- information status
- duplication behaviour
- filetype
- iconstate

Auto DB fields for TCA types

Some ‘management’ fields could already be created automatically based on TCA (uid, pid, language field, tstamp, workspace fields, and so on)

In v12 TCA types category, datetime, slug, json, uuid, file were added

Override in `ext_tables.sql` is possible, but in many cases the file can be removed.

Auto DB fields for TCA types

- file
- email
- check
- folder
- imageManipulation
- language
- group
- flex
- text
- password
- color
- radio
- link
- inline
- number
- select
- input

Default TCA for columns from ‘ctrl’

Some columns are set in the ‘`ctrl`’ section (e.g. `disabled`, `languageField`). These now get a default TCA configuration between processing `Configuration/TCA` and `Configuration/TCA/Overrides`.

They thus can (and are for a few cases) overridden via `Configuration/TCA/Overrides` files.

Schema API

An object oriented API to **read** TCA information.

Types only contain configured fields

Flexforms are parsed, relations are analyzed, and so on.

Meant for use with processing data, not for FormEngine™

DataHandler and Record Factory use it a lot.

Doctrine DBAL driver middlewares

Sits between the generic driver and the actual driver for the database that is used. It can extend the functionality.

There is a `UsableForConnectionInterface` in TYPO3 which allows the middleware to determine if the driver middleware can be used for the particular connection.

Extbase validator messages can be translated

```
#[Extbase\Validate([
  'validator' => 'NotEmpty',
  'options' => [
   'emptyMessage' => 'LLL:EXT:cool_ext/Resources/
Private/Language/
locallang.xlf:validation.coolProperty.notEmpty',
  ],
])]
protected string $coolProperty = '';
```

Extbase validators can access PSR-7 request

Extbase validators which extend AbstractValidator can access the Request object.

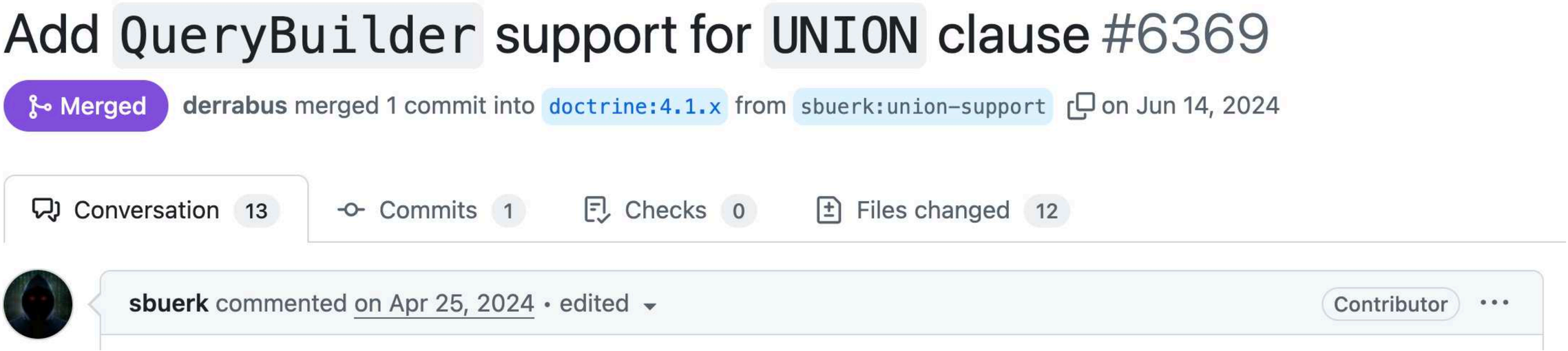
Event to modify available countries

The countryViewHelper uses the country provider to fetch available countries. The new event can modify this list.

Union support in QueryBuilder

```
$queryBuilder->union($firstQuery);
$queryBuilder->addUnion($nextQuery, $unionType);
```

Implemented by adding it to Doctrine!



RecordFactory resolves relations

Create record object from database row with RecordFactory and relations are automatically resolved

New RecordCreationEvent

Before a Record object is made from a database row this event can be used to manipulate fields or even create the entire object.

PSR-14 event AfterRawPageRowPreparedEvent

Can be used to filter, remove, sort items for the BE pagetree.

Example: with news items as page types the “news” pages in the pagetree can be sorted by a date

Prefix for redis cache keys

The option `keyPrefix` allows you to define a prefix for the redis cache keys. This can be used to share redis databases between different caches or between different sites (if the key is unique per cache/site).

Asset collector option ‘external’ stops URL processing

```
$assetCollector->addStyleSheet(‘myCssFile’,
  PathUtility::getAbsoluteWebPath(
    GeneralUtility::getFileAbsFileName(
      ‘EXT:my_ext/Resources/Public/Css/somefile.css’
    )
  ),
  [],
  [‘external’ => true]);

<link rel="stylesheet" href=“/_assets/<hash>/Css/
somefile.css" />
```

Handle file uploads and deletion in Extbase

Model properties of types `FileReference` and `ObjectStorage<FileReference>` can be handled.

```
<f:form.upload />
```

New viewhelper for deleting present file(s).

```
<f:form.uploadDeleteCheckbox />
```

Handle file uploads and deletion in Extbase (2)

Four extbase validators for UploadFile objects

- FileNameValidator
- FileSizeValidator
- MimeTypeValidator
- ImageDimensionsValidator

Handle file uploads and deletion in Extbase (3)

```
#[FileUpload([
  'validation' => [
    'required' => true,
    'maxFiles' => 1,
    'fileSize' => ['minimum' => '0K', 'maximum' =>
'2M'],
    'allowedMimeTypes' => ['image/jpeg', 'image/png'],
  ],
  'uploadFolder' => '1:/user_upload/files/',
])]
protected ?FileReference $file = null;
```

Handle file uploads and deletion in Extbase (4)

For a FileReference property “file”:

```
 <f:form.uploadDeleteCheckbox property=“file”
fileReference=“{object.file}” />
```

For an ObjectStorage<FileReference> property “images”:

```
<f:for each=“{object.images}” as=“oneImage”>
  <f:form.uploadDeleteCheckbox property=“images”
fileReference=“{oneImage}” />
</f:for>
```

Title

Two neighbours on the third f

Images

Delete file

Browse... No files selected.

Country

Belgium

Save

Cache collector API

Page cache tags removed from TSFE

New `frontend.cache.collector` attribute in request to collect tags

`AddCacheTagEvent` for cases where request is not available (internal)

Automatic cache tagging

- table name using `findAll() / countAll()`
- storage pid using `findBy*()`
- records rendered by `CONTENT / RECORDS`
- linked pages (e.g. when target page has endtime)
- linked files (e.g. file metadata changes / file replaced)

GifBuilder supports webp

```
file = GIFBUILDER
file {
  format = webp
  quality = 65
```

Global quality setting: `$TYPO3_CONF_VARS[‘GFX']['webp_quality']`

Quality 101 means “lossless”.

GIFBuilder supports AVIF files

PHP 8.1 has support for this format; now GIFBuilder can use it too. If ImageMagick supports the file format it can be used for output (GraphicsMagick lacks support currently)

CHARACTER SET and COLLATE support per table/column

MySQL and MariaDB support setting `CHARACTER SET` / `COLLATE` per table or column

In `ext_tables.sql` you can set this

Used by `sys_refindex` column ‘`hash`’ to reduce index space

Install Tool allows symlinks for certain directories

- fileadmin
- typo3temp
- var
- user_upload

```
drwx------  26 jigal  staff   832B Jul 27 23:12 _assets
lrwxr-xr-x   1 jigal  staff    14B Aug  1 14:04 fileadmin -> ../filestorage
-rw-------   1 jigal  staff   747B Jul 27 23:11 index.php
drwx------   3 jigal  staff    96B Jul 27 23:12 typo3
drwx------   4 jigal  staff   128B Jul 27 23:12 typo3temp
```

Directory /public/fileadmin permissions mismatch
Default configured permissions are 2775 but current permissions are 0755

File /public/fileadmin/.htaccess
Is a file with the default content and configured permissions of 0664

Directory /public/fileadmin/_temp_
Is a directory with the configured permissions of 2775

Rate limit in felogin password recovery

Maximum 5 attempts in 15 minutes

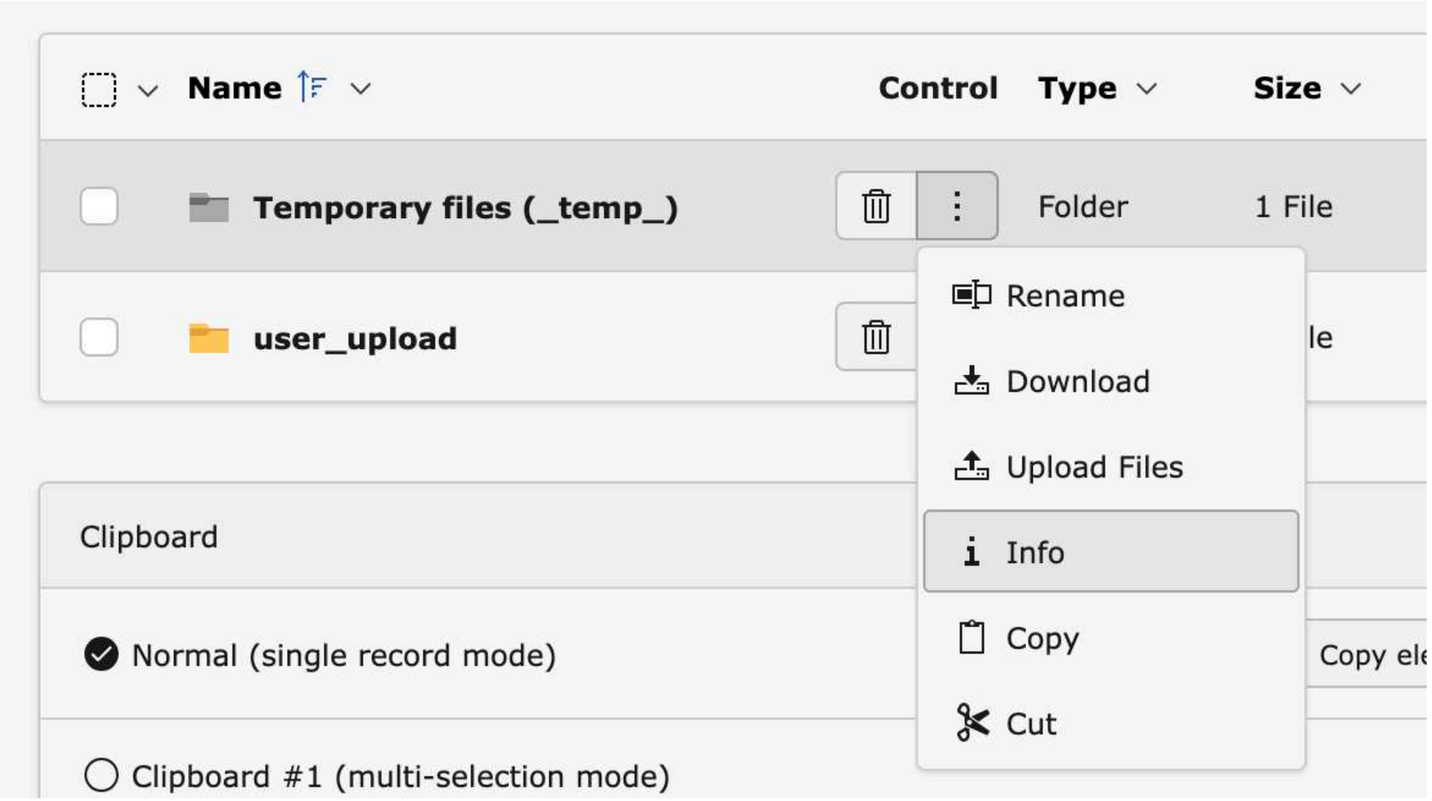
200ms — 3 seconds wait after attempt

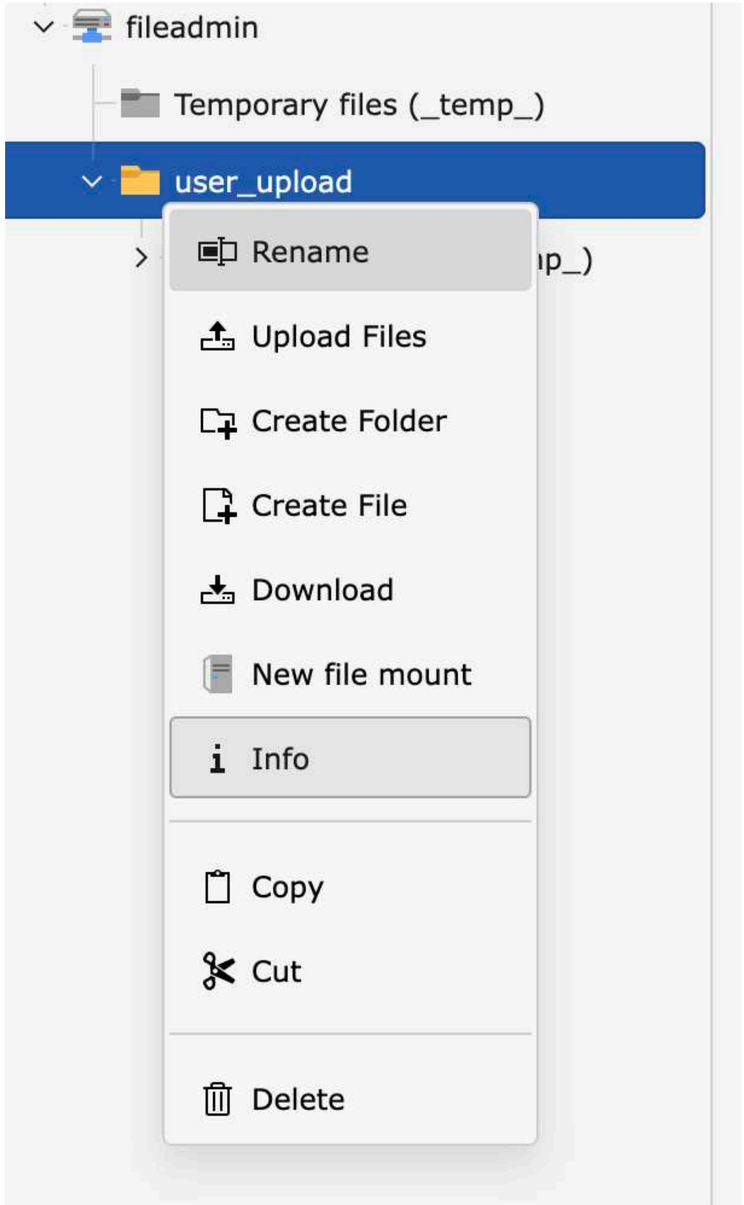
GUI for arrays in global settings

Installation wide settings module has GUI for key-value pairs and lists

Context menu option ‘info’ for folders

Just like for files





Events for Sudo-Mode

- `SudoModeRequiredEvent` before showing the verification dialogue
- `SudoModeVerifyEvent` before verifying the password

Influence step-up authentication process

TYPO3

Thank you!