

TYPO3 Solr as Content Repository

How and why, by Daniel Siepmann

Implemented for Dywidag Systems at web-vision GmbH

Table of Contents

| | | |
|----------|---------------------------------|-----------|
| 1 | Idea / Goal | 4 |
| 2 | Why ext:solr? | 5 |
| 3 | Concept | 6 |
| 4 | Benefits | 7 |
| 5 | Dependencies | 8 |
| 6 | Used TYPO3 core features | 9 |
| 7 | Installation | 10 |
| 8 | Example: News | 11 |
| 8.1 | Custom Page Type | 12 |
| 8.2 | TCA | 13 |
| 8.3 | PageTSConfig | 14 |

| | | |
|----------|-----------------------|-----------|
| 8.4 | TypoScript - Indexing | 15 |
| 8.5 | Searching | 16 |
| 9 | About me | 17 |

1 Idea / Goal

- Use less extensions
- Be more flexible
- Less update hassle, as TYPO3 has no issues, but extensions have
- Less documentation / learning for editors
- Provide awesome filtering, sorting and search for all kind of content
- Use ext:solr as existing solution in combination with TYPO3 Core features

2 Why ext:solr?

- Only available option with all necessary features
- Provides: search
- Provides: facetting
- Provides: indexing
- Provides: Fluid support (through ext:solrfluid)

3 Concept

- Use custom page types instead of extension records, e.g. for News, History, References, ...
- Configure Pages and TCA for editors
- Index everything by ext:Solr and use ext:solrfluid to display and provide necessary features

4 Benefits

- Less components / extensions: Less maintenance
- Just configuration, no programming: Less work & bugs, can be done by integrators
- Future proof: RESTfull, useable for other consumers
- Feature rich: Using atomic design and fluid
- One configuration for everything. Search, news, history, ...

5 Dependencies

- TYPO3, obviously
- ext:solr and ext:solrfluid
- ext:scheduler (for running ext:solr indexing from time to time)
- Apache Solr

6 Used TYPO3 core features

- Custom Page Types
- TCA
- PageTSConfig
- TypoScript
- Fluid

7 Installation

- Nothing to say about TYPO3, right?
- Apache Solr on local development, through Docker:

```
docker run -it -p 8080:8080 \  
-v $(pwd):/opt/solr-tomcat/solr/typo3cores/data \  
writl/solr-typo3
```

- Apache Solr on production, through extension:

```
cd <extdir> && sudo ./install-solr-tomcat.sh
```

- Extensions: install and configure

8 Example: News

| | |
|----------------------------------|-----------|
| 8.1 Custom Page Type | 12 |
| 8.2 TCA | 13 |
| 8.3 PageTSConfig | 14 |
| 8.4 TypoScript - Indexing | 15 |
| 8.5 Searching | 16 |

8.1 Custom Page Type

ext_tables.php:

```
<?php
GLOBALS['PAGES_TYPES'][130] = [
    'type' => 'web',
    'allowedTables' => 'pages_language_overlay,tt_content,...',
    'onlyAllowedTables' => true,
];

\TYPO3\CMS\Core\Utility\ExtensionManagementUtility::addUserTSConfig(
    'options.pageTree.doktypesToShowInNewPageDragArea := addToList(130)
);
```

Also do not forget to add Icon through `\TYPO3\CMS\Core\Imaging\IconRegistry`.

8.2 TCA

- Add select item

```
<?php
\TYPO3\CMS\Core\Utility\ExtensionManagementUtility::addTcaSelectItem
    'pages',
    'doktype',
    [ 'title', 130 ],
    '1',
    'after'
);
```

- Add *types* entry
- Configure fields

8.3 PageTSConfig

Configure Layout:

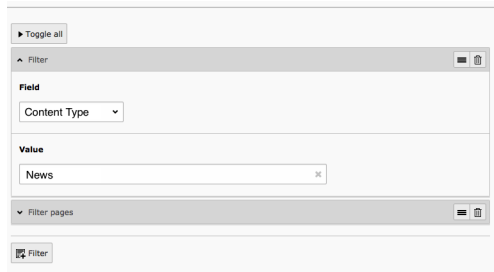
```
mod.web_layout.BackendLayouts.news {
    title = News
    config {
        backend_layout {
            colCount = 1
            rowCount = 1
            rows {
                1 {
                    columns {
                        1 {
                            name = Content
                            colPos = 0
                        }
                    }
                }
            }
        }
    }
}
```

8.4 TypeScript - Indexing

```
plugin.tx_solr.index.queue {
    news < .pages
    news {
        allowedPageTypes = 130
        additionalWhereClause = doktype = 130 AND no_search = 0
        fields {
            pagetype_stringS.value = News
            image_intM = SOLR_RELATION
            image_intM {
                localField = media
                foreignLabelField = uid
                multiValue = 1
            }
        }
        publishDate_tDateS = lastUpdated
    }
}
```

8.5 Searching

Done through flexform. Editor has full control.



The image shows a screenshot of a flexform configuration for a search filter. At the top, there is a button labeled "Toggle all". Below it is a section titled "Filter" with a collapse arrow and a trash icon. Inside this section, there is a "Field" label followed by a dropdown menu set to "Content Type". Below that is a "Value" label followed by a text input field containing the word "News" and a clear button (X). Underneath the "Filter" section is another section titled "Filter pages" with a collapse arrow and a trash icon. At the bottom of the form is a button with a magnifying glass icon and the label "Filter".

9 About me

Daniel Siepmann

TYPO3 Certified Integrator / Developer

TYPO3 since 2009

[@layneobserdia](#)

[daniel-siepmann.de](#)

Blog Post: [TYPO3 everything is content](#)