

# Backend Layouts, Grid Elements, eigene CE's ohne TemplaVoilà

## Backend Layouts

Seit TYPO3 4.5 im Core, ab 6.0 wird auch die standardmäßige Spaltenansicht über ein Default-Backendlayout gerendert.  
-> Implementierung von Drag&Drop einfacher und endlich im Core.

## Backend Layouts

1. Anlegen von Backend-Layouts mit Grid-Wizard im Backend
2. Zuweisung des Ordners als *General Record Storage Pid*
3. Anbindung des Backend-Layouts an Templates für das Frontend

The screenshot displays the TYPO3 Backend interface. On the left is a navigation menu with categories: WEB, FILE, USER TOOLS, ADMIN TOOLS, and HELP. The main area shows a file browser for the 'General Storage Folder' containing 'Backend Layouts'. The right-hand panel is the configuration editor for a 'Backend Layout', featuring an 'Icon' field, a 'Description' text area, and a 'Config' section with a code editor. The code defines a grid layout with 3 columns and 2 rows.

```
backend_layout {
  colCount = 3
  rowCount = 2
  rows {
    1 {
      columns {
        1 {
          name = Header
          colspan = 3
          colPos = 3
        }
      }
    }
    2 {
      columns {
        1 {
          name = Left Content
          colPos = 2
        }
        2 {
          name = Content
          colPos = 0
        }
        3 {
          name = Right Content
          colPos = 1
        }
      }
    }
  }
}
```

Grid wizard

gov6.ak/typo3/sysex/cms/layout/wizard\_backend\_layout.php?&P[fieldConfig][type]=text&P[fieldConfig][rows]=5&P[fi...

The grid wizard interface displays a 3-column layout. The top row is a single wide cell labeled "Name: Header" and "Column number: 3". The bottom row consists of three cells: "Name: Left Content" (Column number: 2), "Name: Content" (Column number: 0), and "Name: Right Content" (Column number: 1). Each cell contains a grey arrow icon pointing towards the right.

Edit Page "Home"

General Access Metadata Appearance Behaviour Resources

**Page Layout**

Frontend Layout Mark as 'New' Until

Layout 2

Backend Layout (this page only) Backend Layout (subpages of this page)

3 Columns  2 Columns

Home

Header

Left Content

Content

Right Content

Aktuelles  
Displaying: LATEST

Das TYPO3 Government Package

Infos in leichter Sprache

TYPO3 für Ämter, Behörden und Gemeinden  
Das TYPO3 Government Package ist eine Musterwebsite für Behörden. Sie dient zur Demonstration und als Starthilfe für barrierefreie Websites mit TYPO3. Eine detaillierte Beschreibung des Pakets, einen Bericht über Hintergründe und Entwicklung, sowie einen Schnellstart finden Sie auf der Seite über das Paket.

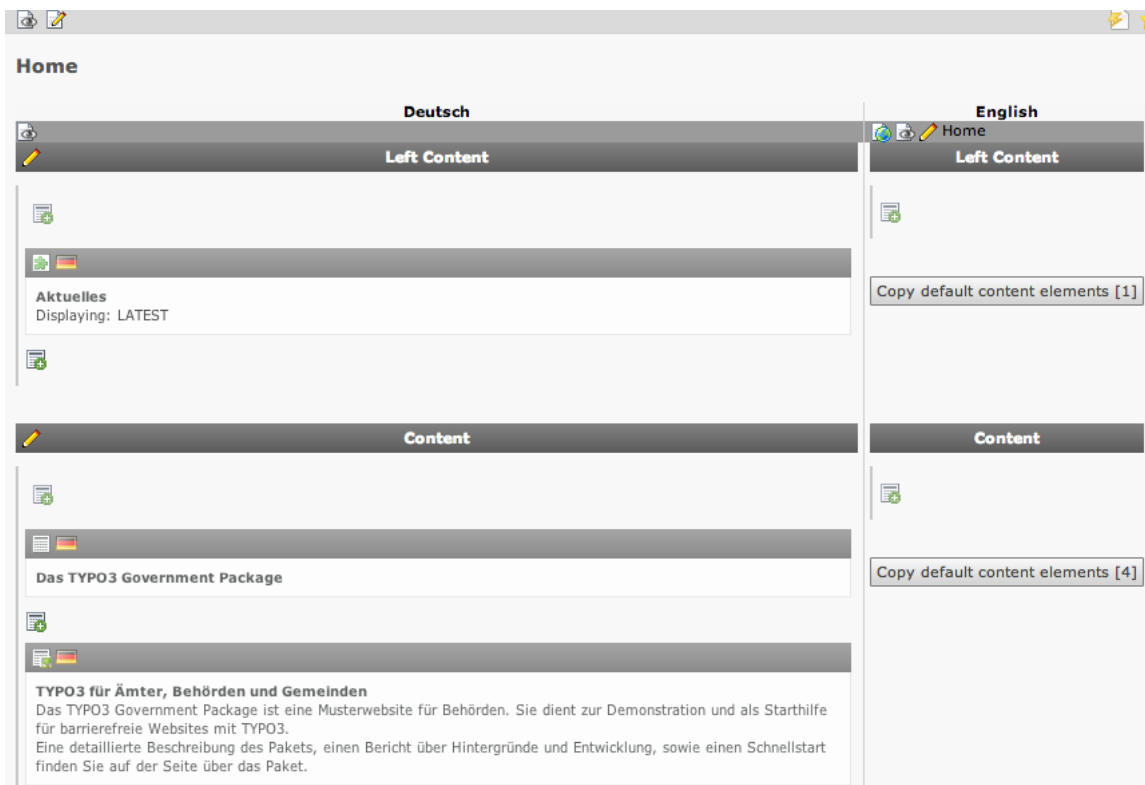
Download und Schnellstart  
Dieses bereits fertig konfigurierte Paket steht Ihnen als Download zur Verfügung. Installieren Sie es in nur wenigen Minuten auf Ihrem Webserver. Testen Sie die Möglichkeiten von TYPO3! Weiter zum Download und zur Schnellstart-Anleitung

Hilfe für Installation und Betrieb  
TYPO3 ist eine Open-Source-Software mit einer großen und aktiven Entwicklergemeinde. Erfahren Sie mehr über Mailinglisten und Internetforen, in denen Sie Hilfe erhalten. In unserem Dienstleisterverzeichnis sind Adressen von auf TYPO3 spezialisierten Profis aufgelistet. Links zu Foren und Dienstleistern

Quellen zum Thema Barrierefreiheit  
Einfach für alle - Einstieg in das Thema Barrierefreiheit  
Text der BITV 2.0  
Aktuelle Bewertung der Verordnung bei Einfach für Alle  
Stand der Umsetzung im Länderrecht  
Aktueller Entwurf der BITV 2.0 in mehreren Sprachen  
Wikipedia-Artikel zur Barrierefreiheit

TYPO3 im Einsatz  
TYPO3 wird bereits von folgenden Behörden und Institutionen verwendet:  
Waldkulturerbe  
Hier wächst Vielfalt  
Gesund ins Leben  
Besser essen und mehr bewegen  
BMELV  
Forschung  
BMELV  
Statistik  
Agrarexportförderung

**Mehrsprachigkeit? Nichts neues...**



## Backend-Layout an das Frontend andocken

```

page = PAGE
page {
    10 = FLUIDTEMPLATE
    10 {
        layoutRootPath = {$layoutsPath}
        file.stdWrap.cObject = CASE
        file.stdWrap.cObject {
            # slide the template
            key.data = levelfield:-1, backend_layout_next_level, slide
            key.override.field = backend_layout
            # default template file
            default = TEXT
            default.value = {$templatesPath}/Pages/index.html
            # template file for backend-layout with ID 2
            2 = TEXT
            2.value = {$templatesPath}/Pages/fullcontent.html
        }
    }
}

```

## Oldschool Template

```

PAGE.10 = TEMPLATE
PAGE.10 {
    template = FILE
    template.file.cObject = CASE
    template.file.cObject {
        key.data = levelfield:-1, backend_layout_next_level, slide
        key.override.field = backend_layout
        key.append = TEXT
        key.append.dataWrap = X{field:layout}

        default = TEXT
        default.value = fileadmin/templates/default.html
        1X0 = TEXT
        1X0.value = fileadmin/templates/be_1_fe_default.html
        1X1 = TEXT
        1X1.value = fileadmin/templates/be_1_fe_1.html
        1X2 = TEXT
        1X2.value = fileadmin/templates/be_1_fe_2.html
        2X0 = TEXT
        2X0.value = fileadmin/templates/be_2_fe_default.html
        2X1 = TEXT
        2X1.value = fileadmin/templates/be_2_fe_1.html
        2X2 = TEXT
        2X2.value = fileadmin/templates/be_2_fe_2.html
    }
}

```

Quelle: <http://lists.typo3.org/pipermail/typo3-team-core/2012-July/052026.html>

### Backend Layouts

- + Bordmittel
- + Freie Wahl der Template-Engine
- Konfiguration kann nicht ausgelagert werden
- +++ Kein Mapping updaten nach Template-Änderung

### Templavoila

- ~ Separate Extension, (betreut von Core-Dev)
- ~ "Alles aus einer Hand"
- ~ Auslagerung soll funktionieren
- ~ Lebensretter sein, wenn z.B. die Templates gelöscht wurden

- Workspaces funktionieren erst seit ein paar Monaten richtig (Änderung von Layouts) +++ Mit richtigen Einstellungen sind Workspaces kein Problem!
- Sprachhandling "unflexibel" +++ Flexibel, getestet, übersichtlich

## Container-Elemente

### Backend Layouts

### Templavoila

nativ vorhanden, beliebige Verschachtelung.  
Vorsicht bei Mehrsprachigkeit und Workspaces!

keine

```
mod.web_templavoilaM1.  
useLiveWorkspaceForReferenceListUpdates = true
```

## Dafür gibt es Grid-Elements

### Alternativen zu Grid Elements

- kb\_nescefe → Entwicklung eingeschlafen, Bug in 4.5 mit Security Token nach über einem Jahr behoben!
- fedext/flux → Viel Magie: Aus Fluidtemplate wird Backend-Ansicht und Content-Element mit Flexform generiert
- multicolumn

## Grid-Elements

### mehr als nur Grids

- Drag&Drop in TYPO3 < 6.0
- kopieren mit Strg-Taste
- neue Elemente anlegen per D&D
- Und bringt auch ein paar neue Bugs mit :-/
- Scheint auch noch einige Probleme mit 6.0 zu haben (Stand vom 22.11)

| ✓                        | #                     | Tracker       | Status         | Priority    | Subject                                                                           | Assignee          | Category                     |
|--------------------------|-----------------------|---------------|----------------|-------------|-----------------------------------------------------------------------------------|-------------------|------------------------------|
| <input type="checkbox"/> | <a href="#">42637</a> | Bug           | Accepted       | Must have   | Gridelements disable alle BE Item previews...                                     | Jo Hasenau        | Layout and UI                |
| <input type="checkbox"/> | <a href="#">42494</a> | Suggestion    | Accepted       | Must have   | Performance: Add index to tt_content:tx_gridelements_container                    | Jo Hasenau        | Frontend rendering           |
| <input type="checkbox"/> | <a href="#">42157</a> | Major Feature | New            | Should have | Native HTML5 drag & drop                                                          | Rocco Georgi      | Layout and UI                |
| <input type="checkbox"/> | <a href="#">42137</a> | Bug           | Under Review   | Must have   | cObj:parentRecordNumber remains 1                                                 | Rik Willems       | Frontend rendering           |
| <input type="checkbox"/> | <a href="#">41937</a> | Bug           | Accepted       | Must have   | Wrong cell insertion for colPos > 255                                             | Jo Hasenau        | Layout and UI                |
| <input type="checkbox"/> | <a href="#">41588</a> | Bug           | Accepted       | Must have   | Errors in documentation                                                           | Anja Leichsenring | Documentation                |
| <input type="checkbox"/> | <a href="#">41494</a> | Task          | On Hold        | Must have   | New extension Grid Elements 2.0 should get own key and branch                     |                   | Extension Management         |
| <input type="checkbox"/> | <a href="#">41486</a> | Suggestion    | Needs Feedback | Should have | Communication? Can I help?                                                        |                   | Documentation                |
| <input type="checkbox"/> | <a href="#">41390</a> | Feature       | Accepted       | Must have   | Dutch translation locallang_db.xml                                                | Rik Willems       | Language and Translation     |
| <input type="checkbox"/> | <a href="#">41342</a> | Feature       | New            | Should have | "Language"-View                                                                   |                   |                              |
| <input type="checkbox"/> | <a href="#">40839</a> | Bug           | Needs Feedback | Should have | Fileupload in Flexform stores absolute filepath                                   | Jo Hasenau        | Flexforms                    |
| <input type="checkbox"/> | <a href="#">40577</a> | Task          | Accepted       | Must have   | Crowd funding and sponsored development - Send me your proposals!                 | Arno Dudek        | Sponsoring and Crowd Funding |
| <input type="checkbox"/> | <a href="#">40006</a> | Bug           | Needs Feedback | Should have | [PATCH] Allow rendering of content using Fluid                                    | Claus Due         | Frontend rendering           |
| <input type="checkbox"/> | <a href="#">39887</a> | Major Feature | Needs Feedback | Should have | Sematic grid by awareness of content                                              | Jo Hasenau        | Frontend rendering           |
| <input type="checkbox"/> | <a href="#">39693</a> | Bug           | Accepted       | Must have   | Exception from tx_gridelements_drawItemHook when reference is deleted             | Jo Hasenau        | Layout and UI                |
| <input type="checkbox"/> | <a href="#">39672</a> | Task          | Accepted       | Should have | flexform typoscript example                                                       | Anja Leichsenring | Documentation                |
| <input type="checkbox"/> | <a href="#">39601</a> | Bug           | Accepted       | Must have   | new_content_el Wizard does not work if TYPO3 is installed in subfolder of webroot | Jo Hasenau        |                              |

### Grid Elements 2.0 - evt. Teil von TYPO3 6.1

startnext  Projekte Pages Was möchtest du finden?

 **Grid Elements 2.0**  
cybercraft  Jetzt Unterwerde

# GRID ELEMENTS 2.0

## Verbessertes TYPO3-Backend mit neuen Features

Start Projektblog **5** Pinnwand **52** Unterstützungen **54**

 **11.530 €**  
von 30.000 € finanziert  **158**  
Fans  **53**  
Supporter  **41**  
Tage noch  **cybercraft**  
Starter

Kein offizielles Core-Projekt?

The screenshot shows the 'Grid Elements 2.0' wizard interface. At the top, there's a 'Columns' dropdown menu. Below it are tabs for 'Typical page content', 'Special elements', 'Form elements', 'Plugins', and 'Grid Elements'. The 'Typical page content' tab is active, showing a grid layout with sections for 'Header', 'Content', and 'Right Content'. The 'Content' section is expanded, showing a 'Deutsch (Default) Grid Element' with a 'New' section containing 'Links' and 'Rechts' options. Below that, there's a 'Deutsch (Default) Text' section with a 'Links' option.

Gleicher Wizard wie bei Backend Layouts

**Edit CE Backend Layout "50-50" on page "Backend Layouts"**

General Configuration Access

**Top Level Layout**  
 Enabled

**Grid Configuration**

```
backend_layout {
  colCount = 2
  rowCount = 1
  rows {
    1 {
      columns {
        1 {
          name = Links
          colPos = 10
        }
        2 {
          name = Rechts
          colPos = 50
        }
      }
    }
  }
}
```

**Content Element Configuration**  
 Flexform Configuration

Flexform Configuration File (overrides direct input)

XML

### Die Features laut Doku (Auszug)

- Comfortable point and click wizard to create backend layout structures
- Flexforms used for configurational stuff only, can be derived from existing data structures
- Original colPos and sorting fields still working
- Top level layouts to exclude certain types of Grid Elements from being used within other Grid Elements
- Drag & drop move and copy actions for the page module
- New content element wizard overlay to drag in new content elements
- Paste icons for pasting copies and references into grid columns
- Completely TypoScript based frontend output

### Wie kriegt man die Ausgabe ins Frontend?

```
# Gridlayout
tt_content.gridelements_pi1.20.10.setup {
  # ID of gridelement
  1 < lib.gridelements.defaultGridSetup
  1 {
    columns {
      # colPos ID
      50 < .default
      50.wrap = <div class="colLeft">|</div>
      # colPos ID
      51 < .default
      51.wrap = <div class="colRight">|</div><div class="clear"> </div>
    }
    wrap = <div class="contentRow">|</div><div class="clear"> </div>
  }
}
```

### Unser Fazit:

Für viele Projekte verwendbar, bei komplexeren Projekten (Multilanguage und Workspaces) werden wir nach wie vor auf Templavoila setzen

### Was packt man in die Backend-Layouts und Grid Elements?

Eigene Content-Elemente

### (Bekannte) Frameworks für eigene Content-Elemente

- templavoila: FCEs funktionieren nicht Standalone, sind nur umständlich versionier-/transportierbar, laut Bugtracker viele Probleme mit 4.7
- fedext/flux/fluidcontent: Alles per Fluid Template konfigurierbar, generiert Flexforms, riesige Sammlung an ViewHelpers. Magic?
- dce - Dynamic Content Elements: Elemente werden im Backend per Wizard angeklickt und in DB gespeichert -> keine Versionierung
- wec\_contentelements: Einfache API, Datei-Struktur muss Konvention verfolgen

## Problem

- zuviel Magie kann und **wird** bei Updates Schwierigkeiten machen
- was passiert, wenn der Autor keine Lust mehr hat?
- wir stellen uns aktuell die Frage, wie wir
  - kb\_nescefe los werden
  - sr\_feuser\_register ersetzen
  - gridelements in 6.0 verwenden
  - wann Templavoila mit 6.0 kompatibel wird
  - Generelles Problem bei Extensions: es kann lange dauern, bis diese mit neuen Versionen verwendet werden können

## Dabei ist das gar nicht so schwer

### ext\_tables.php

```
$pluginSignature = $extensionName . '_pil';
$TCA['tt_content']['columns']['CType']['config']['items'] = array('LLL:EXT:my_ext/Resources/Private/Language/locallang_db.xml:pluginSignature');
$TCA['tt_content']['types'][$pluginSignature]['showitem'] = '
--palette--;LLL:EXT:cms/locallang_ttc.xml:palette.general;general,
header, header_layout,
bodytext;LLL:EXT:cms/locallang_ttc.xml:bodytext_formlabel;;richtext:rte_transform[flag=rte_enabled|mode=ts_css], rte_enabled;L
--div--;LLL:EXT:cms/locallang_ttc.xml:tabs.access, --palette--;LLL:EXT:cms/locallang_ttc.xml:palette.visibility;visibility, --
```

### TypoScript

```
tt_content.myext_pil = FLUIDTEMPLATE
tt_content.myext_pil {
    layoutRootPath = {$layoutsPath}
    file = {$templatesPath}/Content/text.html
}
```

### text.html

```
<f:if condition="{data.header_layout} == 1">
    <h1>{data.header}</h1>
</f:if>
<f:if condition="{data.header_layout} == 2">
    <h2>{data.header}</h2>
</f:if>
<f:format.html>{data.bodytext}</f:format.html>
```

FLUIDTEMPLATE kann auch gegen TEXT, TEMPLATE usw. ausgetauscht werden.

Die Daten sind genau so vorhanden, wie sie aus der Datenbank ausgelesen werden.

→ Ideal geeignet für einfachere Strukturen, dank Fluid sind ViewHelper und einfache Kontroll-Strukturen möglich.

Wann reicht das nicht mehr aus:

v.a. bei komplexeren Eingabe-Möglichkeiten, z.B. IRRE-/DB-Relationen im Content-Element

→ Ausweg: Content-Element durch Extbase schicken und die Möglichkeiten von Models nutzen

### ext\_tables.php



```

$extensionName = 'akaccordion';

\TYPO3\CMS\Extbase\Utility\ExtensionUtility::registerPlugin(
    $_EXTKEY,
    'Accordion',
    'Accordion'
);

\TYPO3\CMS\Core\Utility\ExtensionManagementUtility::addStaticFile($_EXTKEY, 'Configuration/TypoScript', 'Accordion');

$pluginSignature = $extensionName . '_accordion';
$TCA['tt_content']['types'][$pluginSignature]['showitem'] = '
    CType;4;button;1-1-1,
    header,
    header_layout,
    tx_akaccordion_accordion_content,
    --div--;LLL:EXT:cms/locallang_ttc.xml:tabs.access, --palette--;LLL:EXT:cms/locallang_ttc.xml:palette.visibility;visibility, --
    --;LLL:EXT:cms/locallang_ttc.xml:palette.access;access';

```

#### ext\_localconf.php

```

\TYPO3\CMS\Extbase\Utility\ExtensionUtility::configurePlugin(
    'AK.' . $_EXTKEY,
    'Accordion',
    array('Content' => 'accordion'),
    // non-cacheable actions
    array(),
    \TYPO3\CMS\Extbase\Utility\ExtensionUtility::PLUGIN_TYPE_CONTENT_ELEMENT
);

```

#### TypoScript

```

plugin.tx_akaccordion {
    persistence.classes {
        AK\AkAccordion\Domain\Model\TtContent {
            subclasses.akaccordion_accordion = AK\AkAccordion\Domain\Model\Plugins\Accordion
        }
        AK\AkAccordion\Domain\Model\Plugins\Accordion {
            mapping {
                recordType = akaccordion_accordion
                tableName = tt_content
                columns.tx_akaccordion_accordion_content.mapOnProperty = rows
            }
        }
    }
}
config.tx_extbase.persistence.classes {
    AK\AkAccordion\Domain\Model\TtContent {
        mapping.tableName = tt_content
    }
}

```

Basis-Modell für tt\_content kann z.B. aus news-Extension kopiert werden  
Eigene Subclasses erhalten weitere Logik (für Conditions) oder die erweiterten Eigenschaften

```

/**
 * Determine, if the content element has videos
 *
 * @return boolean
 */
public function getHasVideo() {
    if ($this->getVideoMp4() || $this->getVideoOgg() || $this->getVideoWebm()) {
        return TRUE;
    }
    return FALSE;
}

```

#### ak\_accordion/Classes/Domain/Model/Plugins/Accordion.php

```

namespace AK\AkAccordion\Domain\Model\Plugins;
class Accordion extends \AK\AkAccordion\Domain\Model\TtContent {
    /**
     * @var \TYPO3\CMS\Extbase\Persistence\ObjectStorage<AK\AkAccordion\Domain\Model\AccordionContent>
     */
    protected $rows;

    /**
     * @param \TYPO3\CMS\Extbase\Persistence\ObjectStorage $rows
     * @return void
     */
    public function setRows($rows) {
        $this->rows = $rows;
    }

    /**
     * @return \TYPO3\CMS\Extbase\Persistence\ObjectStorage
     */
    public function getRows() {
        return $this->rows;
    }
}

```

#### ak\_accordion/Classes/Controller/ContentController.php

```

namespace AK\AkAccordion\Controller;
class ContentController extends \TYPO3\CMS\Extbase\Mvc\Controller\ActionController {
    /**

```

```

* @return void
*/
public function accordionAction() {
    $data = $this->configurationManager->getContentObject()->data;
    $this->view->assign('content', $this->contentRepository->findByUid($data['uid']));
}

```

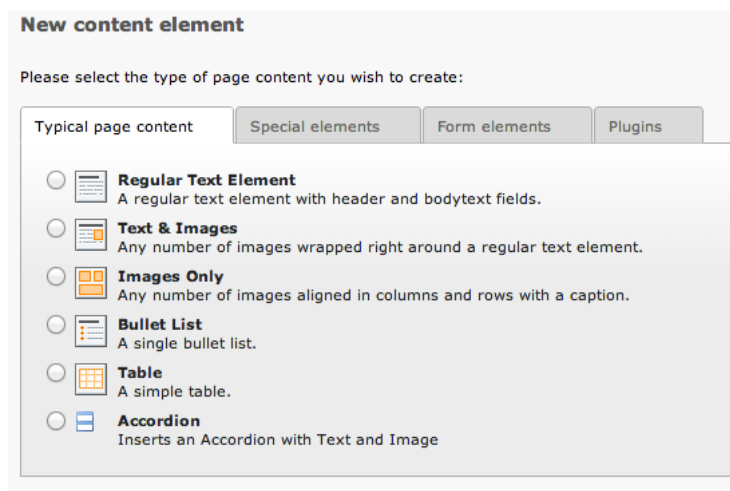
#### ak\_accordion/Resources/Private/Templates/Content/Accordion.html

```

<div class="accordion">
    <f:for each="{content.rows}" as="row">
        <h3>{row.head}</h3>
        <div>
            <f:if condition="{row.image}">
                <div style="float: left; padding-right: 10px;">
                    <f:image src="uploads/tx_akaccordion/{row.image}" alt="" maxWidth="160" maxHeight="80" />
                </div>
            </f:if>
            <f:format.nl2br>{row.bodytext}</f:format.nl2br>
        </div>
    </f:for>
</div>

```

#### Wenn das ganze jetzt noch in den Wizard soll



#### ext\_localconf.php

```

t3lib_extMgm::addPageTSConfig('<INCLUDE_TYPOSCRIPT: source="FILE:EXT:' . $_EXTKEY . '/Configuration/TsConfig/PageTs/wizard.ts">');

```

#### wizard.ts

```

mod.wizards.newContentElement.wizardItems.common.elements.ak_accordion {
    icon = ../typo3conf/ext/ak_accordion/Resources/Public/Icons/Wizard/plugin_accordion.png
    title = LLL:EXT:ak_accordion/Resources/Private/Language/locallang_wizard.xlf:plugin_accordion
    description = LLL:EXT:ak_accordion/Resources/Private/Language/locallang_wizard.xlf:plugin_accordion_description
    tt_content_defValues {
        CType = akaccordion_accordion
    }
}

mod.wizards.newContentElement.wizardItems.common.show := addToList(ak_accordion)

```

#### Hier gibts die Extension:

[https://github.com/akiessling/ak\\_accordion](https://github.com/akiessling/ak_accordion)

#### Weiterführende Links

- <http://fedext.net/>
- <http://blog.mittwald.de/cms/responsive-typo3-3-backend-layouts-und-grid-elements/>
- Der Weg weg von Templavoila  
<http://vimeo.com/41665329>
- Content-Element Wizard: <http://buzz.typo3.org/people/steffen-kamper/article/the-new-content-element-wizard-in-version-43/>