



Contao - wie geht es heute?

Ein Leitfaden für moderne Best Practices in der
Entwicklung mit Contao

Sebastian Zoglowek

- Nickname: zoglo (GitHub, Forum, Slack, überall)
- Fullstack-Entwickler
- Erste Eindrücke seit Contao 4.8 (2019)
- Contao Core-Entwickler seit Februar 2025
- Oveleon GbR
- #Darkmode

Agenda

1. Warum überhaupt?
2. Service Container anstatt \$GLOBALS
 - Service Tagging in Contao
3. Permissions mit Votern
4. Events statt Hooks
5. Rector (und weitere Tools)
5. Rector Regeln schreiben

Warum?

- Symfony seit Contao 4 und seit Jahren etabliert
- Das Rad nicht neu erfinden
- Standards nutzen

Das News-Bundle

```
class ModuleNewsReader extends ModuleNews
{
    public function generate()
    {
        if (TL_MODE == 'BE')
        {
            $objTemplate = new
BackendTemplate('be_wildcard');

            $objTemplate->wildcard = '### NEWS READER
###';
            $objTemplate->title = $this->headline;
            $objTemplate->id = $this->id;
            $objTemplate->link = $this->name;
            $objTemplate->href = 'contao/main.php?
do=themes&table=tl_module&act=edit&id=' . $this-
>id;

            return $objTemplate->parse();
        }

        // Return if no news item has been specified
        if (!$this->Input->get('items'))
        {
            return '';
        }

        $this->news_archives = $this-
>sortOutProtected(deserialize($this-
>news_archives));

        // Return if there are no archives
        if (!is_array($this->news_archives) ||
count($this->news_archives) < 1)
        {
            return '';
        }

        return parent::generate();
    }
}
```

Das News-Bundle (als schlechtes Beispiel)

Commit 4a2809d

 leofeyer committed on Aug 9, 2008

Initial commit

 3.5 ·  3.5.40 ... 2.6.0

-  moduleNewsList.php
-  ModuleNewsMenu.php
-  **ModuleNewsReader.php**
-  News.php
- ▼  config
 -  .htaccess
 -  config.php
 -  database.sql
- ▼  dca
 -  .htaccess

 **Some content is hidden**
Large Commits have some content hidden by default. Use the search bar to find more.

 **1,571 files changed** +212762 -0 lines changed

▼ **INSTALL.txt** 

...	@@ -0,0 +1,56 @@
1	+ =====
2	+ TYPOlight web content management system
3	+ =====
4	+

Der erste commit

Best Practices - seit 2016

2016 - Contao 4 Extension Development

(David Greminger)

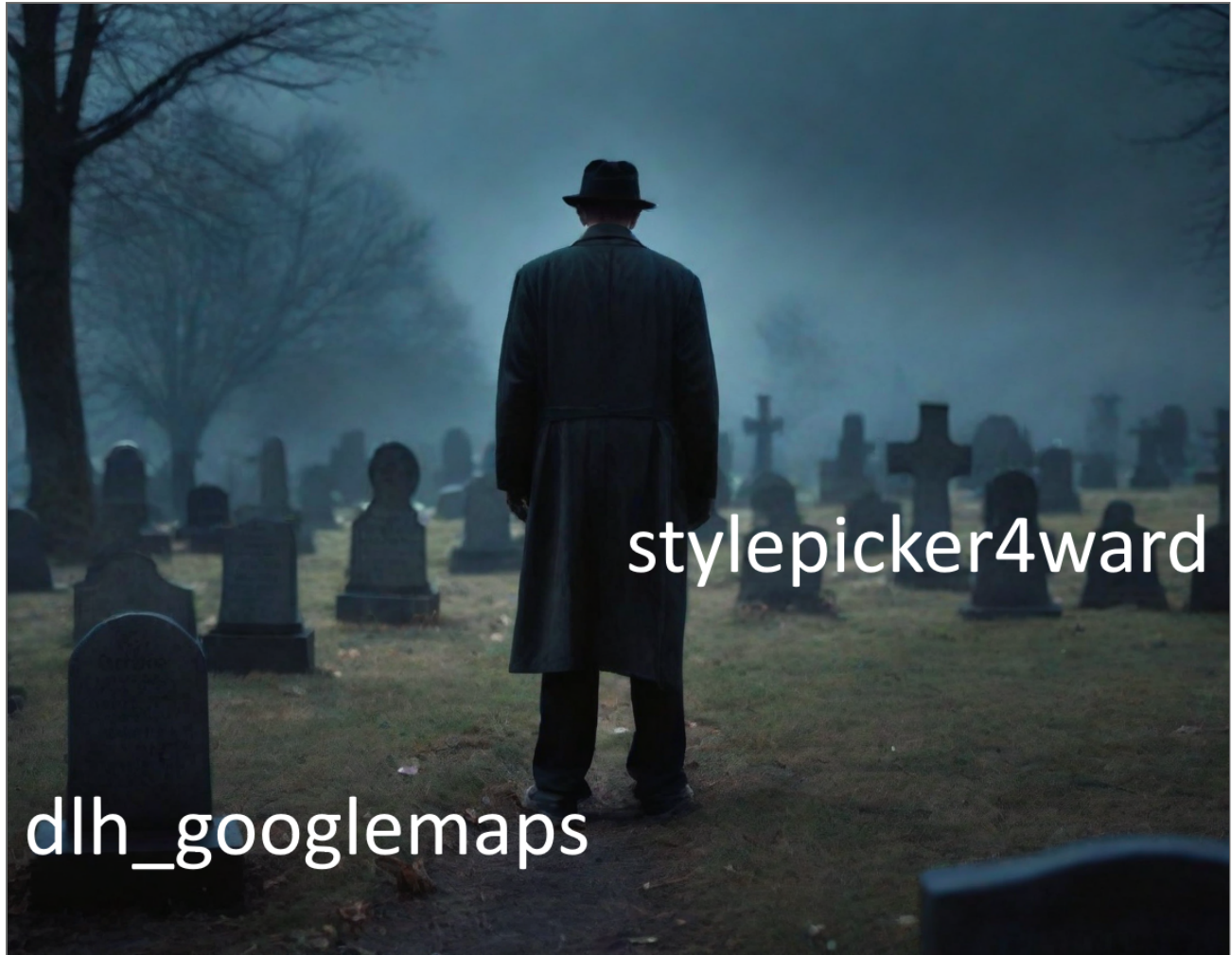
Video zum Konferenzvortrag

2017 - Extension from scratch in Contao 4

(Christian Schiffler)

Video zum Konferenzvortrag

Zwei verwaiste Plugins als Beispiel



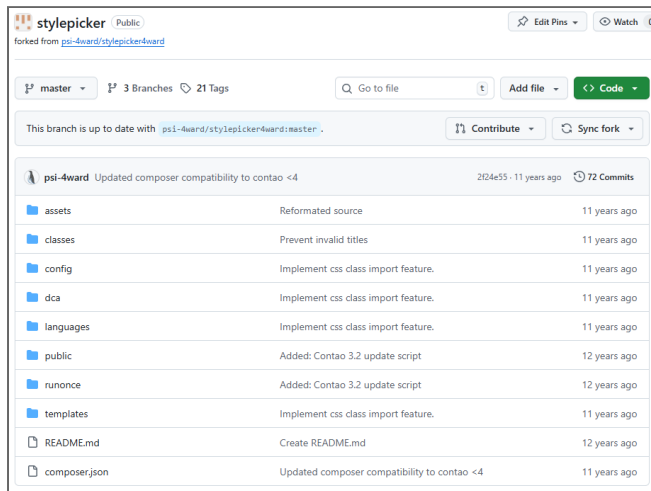
Verfügbar?

Klar!

<https://github.com/contao-graveyard/stylepicker>

<https://github.com/contao-graveyard/googlemaps>

Stylepicker4ward



stylepicker Public
forked from psi-4ward/stylepicker4ward

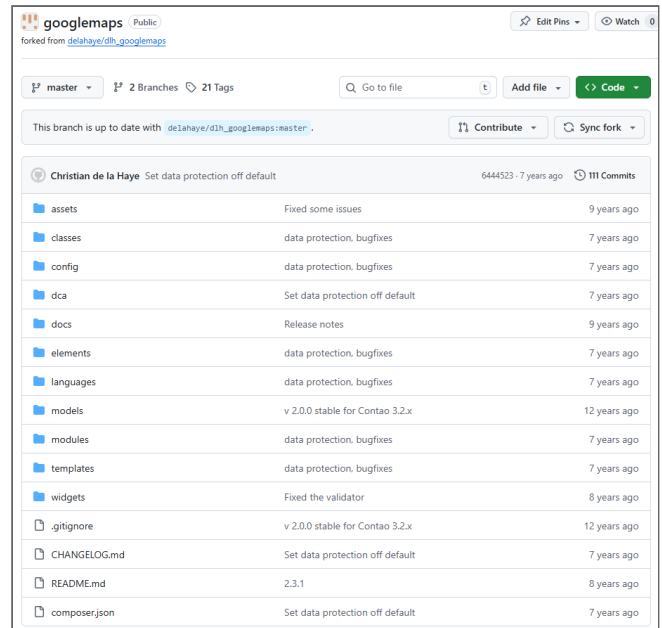
master 3 Branches 21 Tags

Go to file Add file Code

This branch is up to date with psi-4ward/stylepicker4ward:master. Contribute Sync fork

psi-4ward	Updated composer compatibility to contao <4	2f24e55 · 11 years ago	72 Commits
assets	Reformatted source	11 years ago	
classes	Prevent invalid titles	11 years ago	
config	Implement css class import feature.	11 years ago	
dca	Implement css class import feature.	11 years ago	
languages	Implement css class import feature.	11 years ago	
public	Added: Contao 3.2 update script	12 years ago	
runonce	Added: Contao 3.2 update script	12 years ago	
templates	Implement css class import feature.	11 years ago	
README.md	Create README.md	12 years ago	
composer.json	Updated composer compatibility to contao <4	11 years ago	

dlh_googlemaps



googlemaps Public
forked from delahaye/dlh_googlemaps

master 2 Branches 21 Tags

Go to file Add file Code

This branch is up to date with delahaye/dlh_googlemaps:master. Contribute Sync fork

Christian de la Haye Set data protection off default 6444523 · 7 years ago 111 Commits

assets	Fixed some issues	9 years ago
classes	data protection, bugfixes	7 years ago
config	data protection, bugfixes	7 years ago
dca	Set data protection off default	7 years ago
docs	Release notes	9 years ago
elements	data protection, bugfixes	7 years ago
languages	data protection, bugfixes	7 years ago
models	v 2.0.0 stable for Contao 3.2.x	12 years ago
modules	data protection, bugfixes	7 years ago
templates	data protection, bugfixes	7 years ago
widgets	Fixed the validator	8 years ago
.gitignore	v 2.0.0 stable for Contao 3.2.x	12 years ago
CHANGELOG.md	Set data protection off default	7 years ago
README.md	2.3.1	8 years ago
composer.json	Set data protection off default	7 years ago

Manual installation: copy all files to system/modules/

Step 1

BLIND UMSCHREIBEN IN EIN CONTAO-MANAGER-BUNDLE

Step 2: Use Rector



```

<input type="hidden" name="FORM_SUBMIT" value="li_submit">
<input type="hidden" name="REQUEST_TOKEN" value="REQUEST_TOKEN">
<input type="hidden" name="REQUEST_TOKEN" value="System::getContainer()->get('contao.csrf.token_manager')->getDefaultTokenValue().">
<input type="hidden" name="MAX_FILE_SIZE" value="1.060843[7c_c0f1d]" maxfilesize="1">

<div class="li_thco">
    @@
    <div class="li_formbody_submit">
        <input class="li_submit_container">
        <input type="submit" name="save" id="save" class="li_submit" accesskey="s" value="specialchars{$0.00A3[7c_c0f1d]}%BC["li_submit"]{0}"/>
        <input type="submit" name="save" id="save" class="li_submit" accesskey="s" value="stringtailspecialchars{$0.00A3[7c_c0f1d]}%BC["li_submit"]{0}"/>
    </div>

</div>

----- end diff -----

Applied rules:
+ ConstantServiceCollector
+ LongArrayToShortArrayFactor
+ ChangeSwitchToFactorFactor
+ NullToIntegerIntegerFunctionCollector
+ RenameClassFactor
+ FunctionCallToStaticCollector

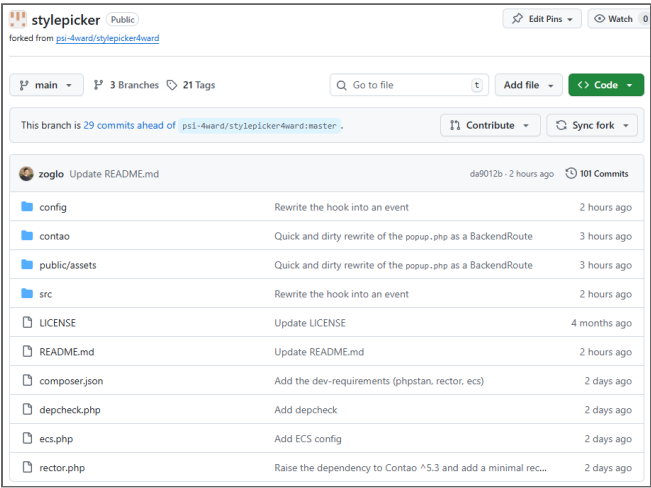
[OK] 26 files have been changed by Rector

```

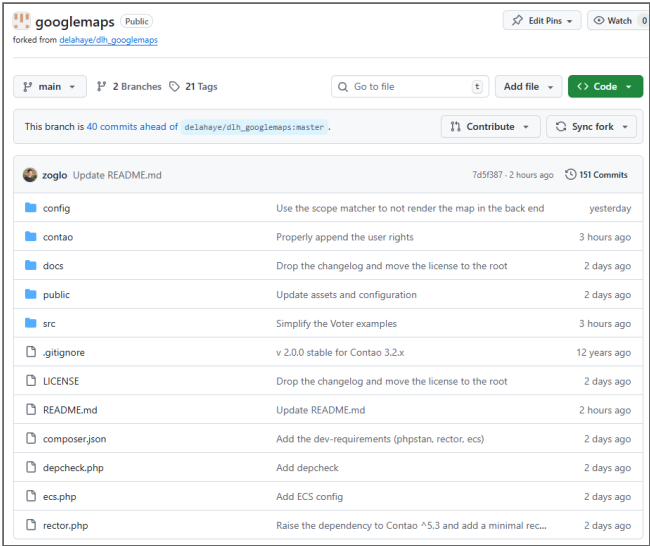
Zack feddig

"Re(fa)ctored"

Stylepicker4ward

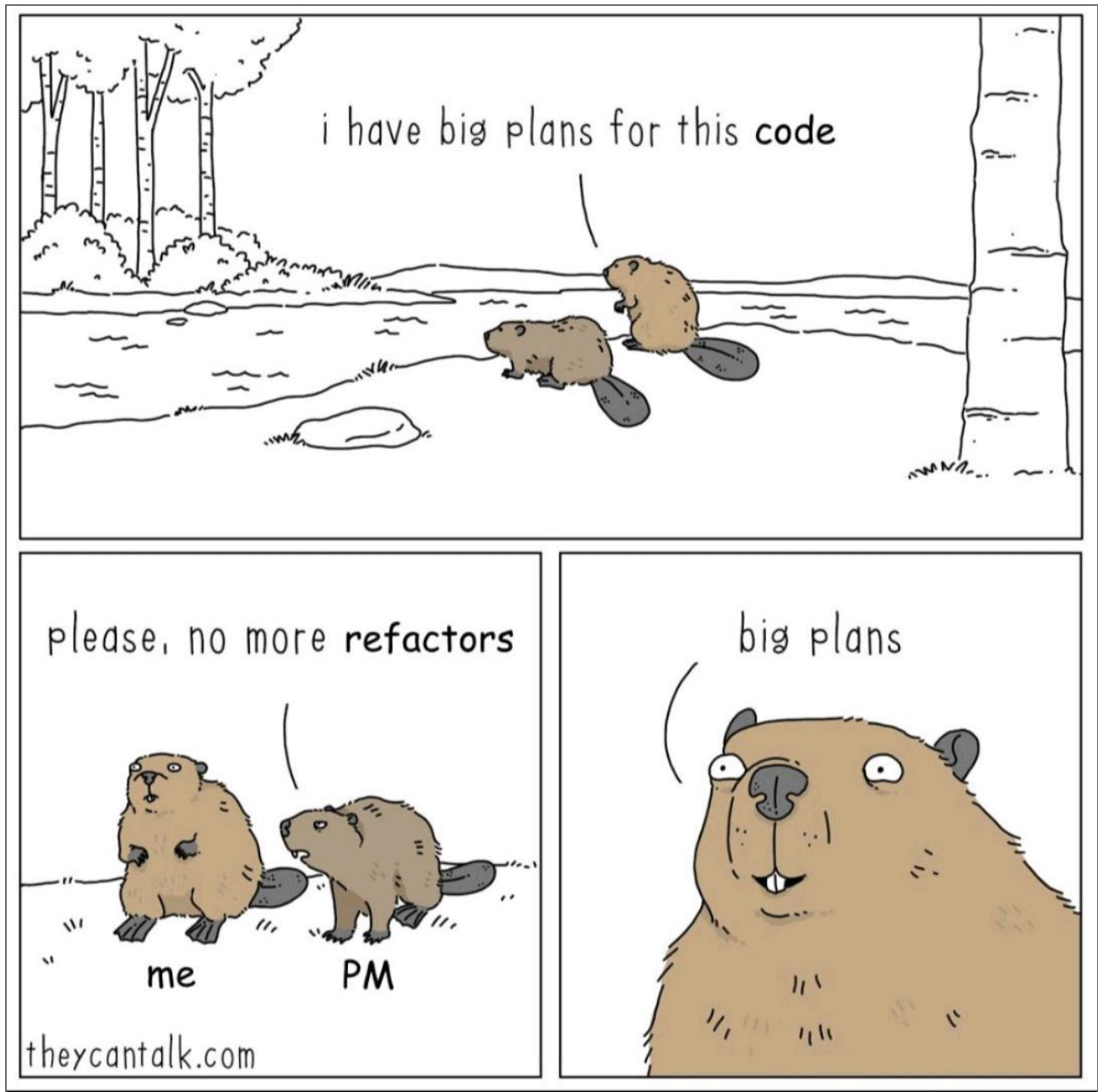


dlh_googlemaps



Vortrag zu Ende

Vortrag zu Ende ... ?



Service Container anstatt \$GLOBALS

\$GLOBALS

Teil der Superglobals - interne Variablen, die überall verfügbar sind

Assoziatives Array, das Referenzen auf alle Variablen enthält

Service Container

Ein PHP-Objekt, welches für die Instanziierung von Objekten bzw. Services zuständig ist.

Es weiß über seine Abhängigkeiten Bescheid, da alles (vor-)konfiguriert

[https://symfony.com/doc/current/components/depend](https://symfony.com/doc/current/components/dependency_injection.html)

Services

- PHP Objekte
- Stateless
- Sollen austauschbar sein
- Werden konfiguriert in der "services.yaml"
- Per Default nur eine Instanz / Singleton
- Kann man testen

Database::getInstance()

```
class Database
{
    protected function __construct()
    {
        $this->resConnection = System::getContainer()-
>get('database_connection');

        if (!\is_object($this->resConnection)) {
            throw new \Exception(\sprintf('Could not
connect to database (%s)', $this->error));
        }
    }

    public static function getInstance()
    {
        return static::$objInstance ??= new static();
    }
}
```

Vorher

```
$objContent = Database::getInstance()  
->prepare('SELECT type,pid FROM tl_content WHERE  
id=?')  
->execute($id)  
;  
  
$id = $objContent->pid;  
$cond = $objContent->type;
```

Nachher

```
class StylePickerController extends
AbstractBackendController
{
    public function __construct(
        private readonly Connection $connection //
        Datenbank Service
    ) {}

    // ...
    $objContent = $this->connection-
>fetchAssociative(
    'SELECT type, pid FROM tl_content WHERE id =
?', [$id]
    );

    $id = $objContent['pid'] ?? null;
    $condition = $objContent['type'] ?? null;
```

Service-Tagging

und wie das Contao CMS es für Entwickler vereinfacht

Beispiel eines Content-Elements

Warum sollte ich meine Inhaltselemente umschreiben?

- Fragment Controller
- Seit Symfony 2.2 (aber auch in 2.0)
- Fragmente = Schnipsel
- Caching von Schnipseln auf der Page (ESI)
- Mit Twig viel mehr möglich :)

Services müssen in der services.yaml definiert werden

```
services:
  _defaults:
    autoconfigure: true

ContaoGraveyard\GoogleMapsBundle\Controller\ContentElement
\GoogleMapsController:
  tags:
    -
      type: dlh_googlemaps
      name: contao.content_element
      category: media
      template: ce_dlh_googlemaps_default
```

Geht das nicht einfacher?

Attributes!

(früher auch über Annotations)

AsFrontendModule

AsContentElement

AsPage

AsPickerProvider

AsCronJob

AsHook

AsInsertTag

AsBlockInsertTag

AsInsertTagFlag

AsCallback

#[AsContentElement]

```
#[AsContentElement(  
    type: 'dlh_googlemaps',  
    category: 'media',  
    template: 'ce_dlh_googlemaps_default',  
)]  
class GoogleMapsController extends  
AbstractContentElementController  
{
```

Und nun?

```
services:
  _defaults:
    autoconfigure: true

#ContaoGraveyard\GoogleMapsBundle\Controller\ContentElement\GoogleMapsController: ~
...ContentElement\GoogleMapsController: ~

# Yes, it's invalid, I know
```

Aber was ist mit Autowiring!?



Klar, kannst du machen!

```
services:
  _defaults:
    autoconfigure: true
    autowire: true

  ContaoGraveyard\GoogleMapsBundle\:
    resource: '../src/'
    exclude: '../src/{Event,Model,Whatever.php}'
```

Aber nicht in public bundles!

Mr. Codestyle wants to talk to you

Closed

Added access voters for models in front end #1639
aschempp wants to merge 14 commits into [contao:5.x](#) from [aschempp:feature/model-vote...](#)

core-bundle/src/Resources/config/voters.yml (Outdated)

```
1 + services:
2 +   _defaults:
3 +     autoconfigure: true
4 +     autowire: true
```

 bytehead on Jun 25, 2020

Since when do we use autowiring?



Member

...

 leofeyer on Jun 25, 2020



Member

...

  1

Public Bundles kennen die Services in deinem Container nicht (in der Theorie)

In der App hat man Kontrolle über alles

Permissions und Voter

Warum?

- Legacygründe: man hat immer alle Berechtigungen
- Und verweigert danach
- Nicht mehr selber Berechtigungen schreiben
- Lass es das Framework machen
- C R U D

Wie ging es damals?

DO NOT CHANGE THIS ORDER!

```
/**
 * Initialize the controller.
 *
 * 1. Import user
 * 2. Call parent constructor
 * 3. Authenticate user
 * 4. Load language files
 * DO NOT CHANGE THIS ORDER!
 */
public function __construct()
{
    $this->import(BackendUser::class, 'User');
    $this->import('Database');

    $this->User->authenticate();

    $this->loadLanguageFile('default');
    $this->loadLanguageFile('modules');
}
```

onLoadCallback!

```

$GLOBALS['TL_DCA']['tl_dlh_googlemaps'] = [
    'config' => [
        // ...
        'onload_callback' => [
            ['tl_dlh_googlemaps', 'checkPermission'],
        ],
    ],
    // ...
];

public function checkPermission(): void
{
    if ($this->User->isAdmin) {
        return;
    }
    // Set root IDs
    if (!is_array($this->User->dlh_googlemapss) || $this->User
[[]]) {
        $root = [0];
    } else {
        $root = $this->User->dlh_googlemapss;
    }
    $GLOBALS['TL_DCA']['tl_dlh_googlemaps']['list']['sorting']
    // Check permissions to add Maps
    if (!$this->User->hasAccess('create', 'dlh_googlemapsp'))
    {
        $GLOBALS['TL_DCA']['tl_dlh_googlemaps']['config']['clc
    }
}

```

```
// Check current action
switch (Input::get('act')) {
    case 'create':
    case 'select':
        // Allow
        break;
```

```

    case 'edit':
        // Dynamically add the record to the user profile
        if (!in_array(Input::get('id'), $root, true)) {
            $arrNew = $this->Session->get('new_records');
            if (is_array($arrNew['tl_dlh_googlemaps']) &&
in_array(Input::get('id'), $arrNew['tl_dlh_googlemaps'], true)
                // Viel Code um Berechtigungen auf Gruppenebene
                // ...
            }
        }
        // no break;
    case 'copy':
    case 'delete':
    case 'show':
        if (!in_array(Input::get('id'), $root, true) || (Input
'delete' && !$this->User->hasAccess('delete', 'dlh_googlemaps'
            System::getContainer()->get('monolog.logger.contao
                LogLevel::ERROR,
                'Not enough permissions to ' . Input::get('act
Input::get('id')) . '"',
            [
                'contao' => new ContaoContext(__METHOD__,
ContaoContext::ERROR),
            ],
        ));
        $this->redirect('contao/main.php?act=error');
    }
    break;

```

```

case 'editAll':
case 'deleteAll':
case 'overrideAll':
    $session = $this->Session->getData();
    if (Input::get('act') === 'deleteAll' && !$this->User-
'dlh_googlemapsp')) {
        $session['CURRENT']['IDS'] = [];
    } else {
        $session['CURRENT']['IDS'] = array_intersect($sess
['IDS'], $root);
    }
    $this->Session->setData($session);
    break;

default:
    if ((string) Input::get('act') !== '') {
        System::getContainer()->get('monolog.logger.contac
LogLevel::ERROR,
        'Not enough permissions to ' . Input::get('act
        [
            'contao' => new ContaoContext(__METHOD__,
ContaoContext::ERROR),
        ],
    );
    $this->redirect('contao/main.php?act=error');
    }
    break;
}

```

Voters

This feature is available in Contao **4.7** and later.

This feature is available in Contao **4.10** and later.

This feature is available in Contao **4.12** and later.

This feature is available in Contao **5.0** and later.

Jetzt aber!

```

class GoogleMapsAccessVoter extends AbstractDataContainerVoter
{
    public function __construct(private readonly AccessDecisionManager
$accessDecisionManager)
    {}

    protected function getTable(): string
    {
        return 'tl_dlh_googlemaps';
    }

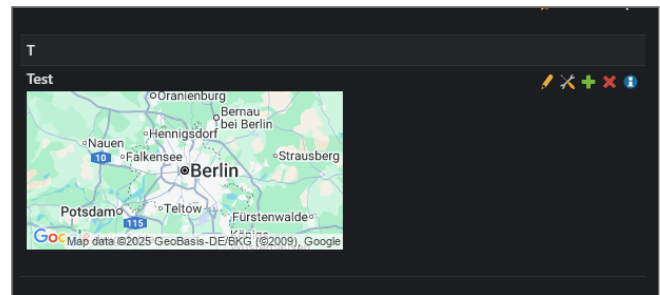
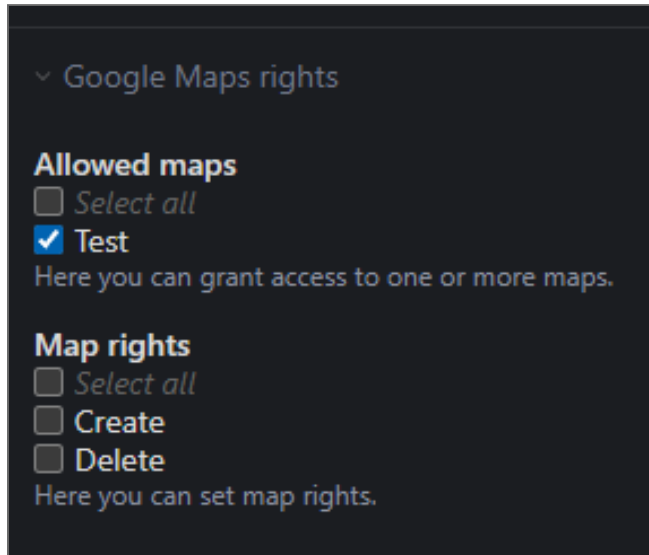
    protected function hasAccess(TokenInterface $token,
CreateAction|DeleteAction|ReadAction|UpdateAction $action): bool
    {
        if (!$this->accessDecisionManager->decide($token,
[ContaoCorePermissions::USER_CAN_ACCESS_MODULE . '.dlh_googlemaps']))
            return false;
        }

        return match (true) {
            $action instanceof CreateAction => $this->accessDecisionManager
['contao_user.dlh_googlemapsp'], 'create'),
            $action instanceof ReadAction,
            $action instanceof UpdateAction => $this->accessDecisionManager
['contao_user.dlh_googlemapss'], (int) $action->getCurrentId()),
            $action instanceof DeleteAction => $this->accessDecisionManager
['contao_user.dlh_googlemapss'], (int) $action->getCurrentId())
            && $this->accessDecisionManager->decide($token,
['contao_user.dlh_googlemapsp'], 'delete'),
        };
    }
}

```

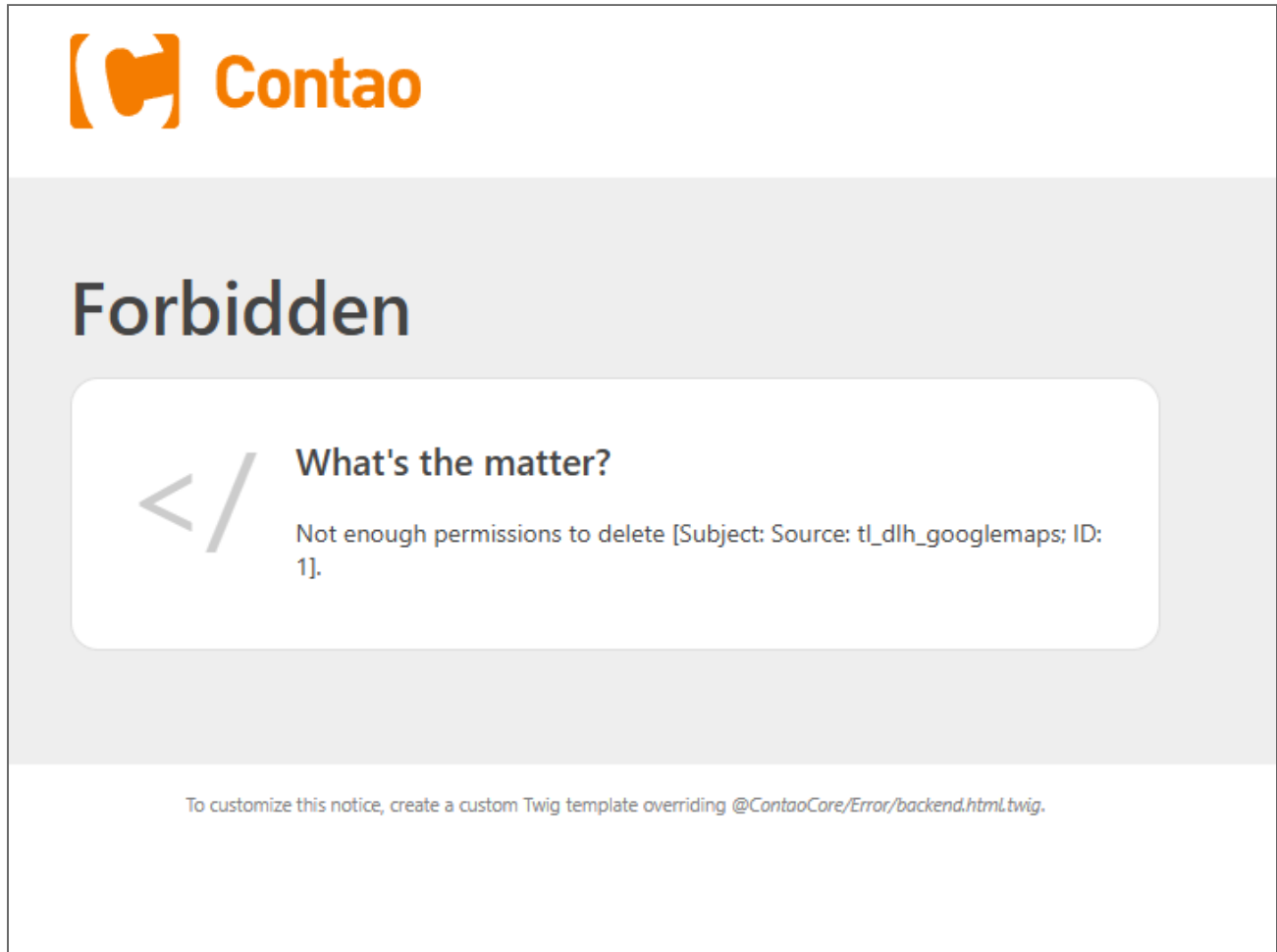
Permissions mit CRUD erledigt

Jetzt erstmal löschen!



Oh... Forbidden

Es fehlen die Button-Callbacks ...



Dafür gibt es jetzt den OperationsBuilder

```

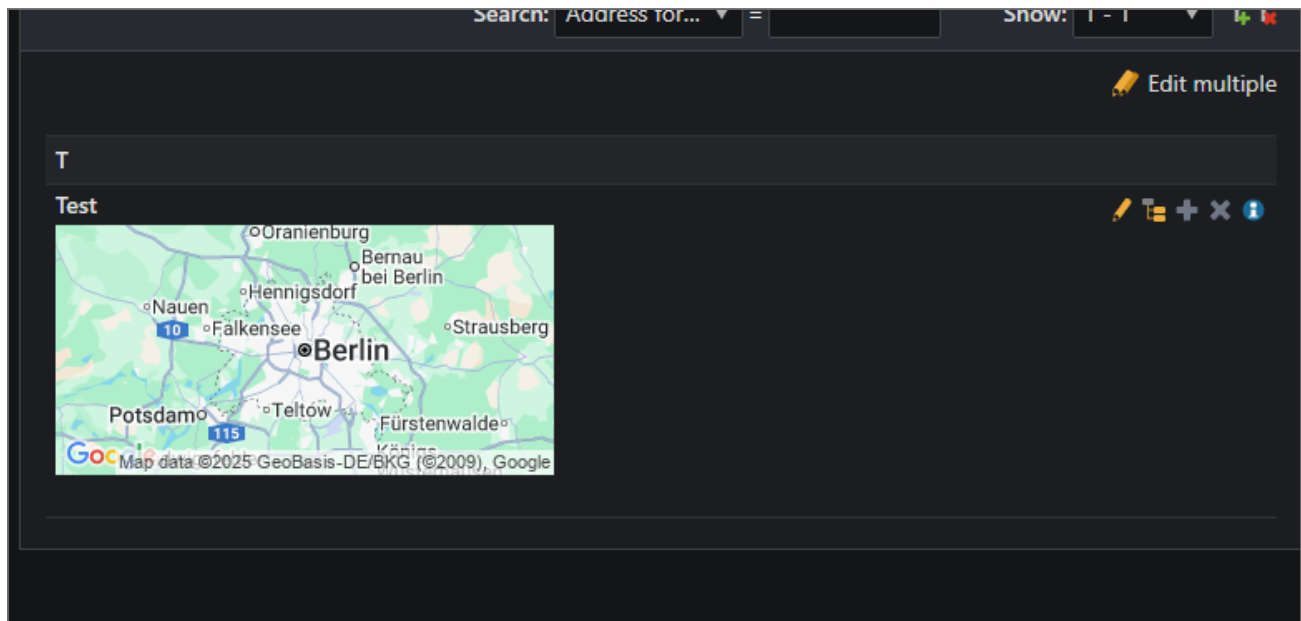
],
'global_operations' => [
    'all' => [
        'label' => $GLOBALS['TL_LANG']['MSC']['all'],
        'href' => 'act=select',
        'class' => 'header_edit_all',
        'attributes' => 'onclick="Backend.getScrollOffset();" accesskey="e"',
    ],
],
'operations' => [
    'edit' => [
        'label' => $GLOBALS['TL_LANG']['tl_dlh_googlemaps']['edit'],
        'href' => 'table=tl_dlh_googlemaps_elements',
        'icon' => 'edit.gif',
        'attributes' => 'class="contextmenu"',
    ],
    'editheader' => [
        'label' => $GLOBALS['TL_LANG']['tl_dlh_googlemaps']['editheader'],
        'href' => 'act=edit',
        'icon' => 'header.gif',
        'button_callback' => ['tl_dlh_googlemaps', 'editHeader'],
        'attributes' => 'class="edit-header"',
    ],
    'copy' => [
        'label' => $GLOBALS['TL_LANG']['tl_dlh_googlemaps']['copy'],
        'href' => 'act=copy',
        'icon' => 'copy.gif',
    ],
    'delete' => [
        'label' => $GLOBALS['TL_LANG']['tl_dlh_googlemaps']['delete'],
        'href' => 'act=delete',
        'icon' => 'delete.gif',
        'attributes' => 'onclick="if (confirm(\'\' . ($GLOBALS['TL_LANG']['MSC']['deleteConfirm']
    ],
    'show' => [
        'label' => $GLOBALS['TL_LANG']['tl_dlh_googlemaps']['show'],
        'href' => 'act=show',
    ],
],
],
'global_operations' => [
    'all' => [
        'label' => $GLOBALS['TL_LANG']['MSC']['all'],
        'href' => 'act=select',
        'class' => 'header_edit_all',
        'attributes' => 'onclick="Backend.getScrollOffset();" accesskey="e"',
    ],
],
'operations' => [
    'edit' => [
        'label' => $GLOBALS['TL_LANG']['tl_dlh_googlemaps']['edit'],
        'href' => 'table=tl_dlh_googlemaps_elements',
        'icon' => 'edit.gif',
        'attributes' => 'class="contextmenu"',
    ],
    'editheader' => [
        'label' => $GLOBALS['TL_LANG']['tl_dlh_googlemaps']['editheader'],
        'href' => 'act=edit',
        'icon' => 'header.gif',
        'button_callback' => ['tl_dlh_googlemaps', 'editHeader'],
        'attributes' => 'class="edit-header"',
    ],
    'copy' => [
        'label' => $GLOBALS['TL_LANG']['tl_dlh_googlemaps']['copy'],
        'href' => 'act=copy',
        'icon' => 'copy.gif',
    ],
    'delete' => [
        'label' => $GLOBALS['TL_LANG']['tl_dlh_googlemaps']['delete'],
        'href' => 'act=delete',
        'icon' => 'delete.gif',
        'attributes' => 'onclick="if (confirm(\'\' . ($GLOBALS['TL_LANG']['MSC']['deleteConfirm']
    ],
    'show' => [
        'label' => $GLOBALS['TL_LANG']['tl_dlh_googlemaps']['show'],
        'href' => 'act=show',
    ],
],
],
// Palettes
'palettes' => [
    '__selector__' => [
        'useMapTypeControl',
        'useZoomControl',
        'usePanControl',
        'useRotateControl',
        'useScaleControl',
        'useStreetViewControl',
        'useOverviewMapControl',
        'useClusterer',
    ],
    'default' => '{title_legend},title,geocoderAddress,geocoderCountry,center,mapSize,zoom,{maptyp
],
// Subpalettes
'subpalettes' => [
    'useMapTypeControl' => 'mapTypeControlStyle,mapTypeControlPos',
    'useZoomControl' => 'zoomControlStyle,zoomControlPos',
    'useRotateControl' => 'rotateControlStyle,rotateControlPos',
    'usePanControl' => 'panControlStyle,panControlPos',
    'useScaleControl' => 'scaleControlPos',
    'useStreetViewControl' => 'streetViewControlPos',
    'useClusterer' => 'clustererMap'
]

```

```
'label' => [  
    'fields' => ['title'],  
    'format' => '%s',  
    'label_callback' => ['tl_dlh_googlemaps', 'listRecords'],  
],  
'global_operations' => [  
    'all',  
],  
'operations' => [  
    'edit',  
    'children',  
    'copy',  
    'delete',  
    'show',  
],  
],
```

* Seit Contao 5.5 kann man die Defaults auch weglassen

Kaum löscht man paar Zeilen Code
Hat man das Feature implementiert



Events statt Hooks!

Aber warum?

- Hooks greifen auf Globals zu (deprecated)
- ... ich, Ich, ICH! ICH MUSS ZUERST! (Priority)
- Events kannst du "decoraten"! (überschreiben)
- Events sind Objekte

Als Hook

```
if (isset($GLOBALS['TL_HOOKS']  
['stylepicker4ward_getFilter']) &&  
is_array($GLOBALS['TL_HOOKS']  
['stylepicker4ward_getFilter'])) {  
    foreach ($GLOBALS['TL_HOOKS']  
['stylepicker4ward_getFilter'] as $callback) {  
        System::importStatic($callback[0]);  
        $result = $this->{$callback[0]}->{$callback[1]}  
($table, $id);  
        if (is_array($result)) {  
            [$table, $layout, $section, $condition] =  
$result;  
            break;  
        }  
    }  
}
```

Als Event

```
$event = new GetStylePickerFilterEvent($table, (int)
$id);
$this->eventDispatcher->dispatch($event);

$layout = $event->getLayout();
$section = $event->getSection();
$condition = $event->getCondition();
```

Und darauf kannst du hören (EventListener)

```
#[AsEventListener]
class StylePickerEventListener
{
    public function __invoke(GetStylePickerFilterEvent $event): void
    {
        if ($event->getTable() !== 'my_table') {
            return;
        }

        $event->setLayout(213);
        $event->setCondition('foobar');
        $event->setSection('your_section');
    }
}
```

Rector (und andere Tools)

Übersicht

	Meldet Fehler	Fixed Fehler
Coding standard	PHP_Codesniffer	PHP-CS-Fixer, ECS
Logik	PHPStan, Psalm	Rector

CODING STANDARD

Modifiziert Code und ist Token-Basiert (Lexing und Tokenisierung)
PHPToken oder `token\_get\_all`

```
$source = <<<'code'
<!--?php

class A
{
    const PUBLIC = 1;
}
code;

$tokens = token_get_all($source, TOKEN_PARSE);
foreach ($tokens as $token) {
    if (is_array($token)) {
        echo token_name($token[0]) , PHP_EOL;
    }
}
?-->
```

```
T_OPEN_TAG  
T_WHITESPACE  
T_CLASS  
T_WHITESPACE  
T_STRING  
T_CONST  
T_WHITESPACE  
T_STRING  
T_LNUMBER
```

STATIC ANALYZERS

Basierend auf dem AST (Abstract Syntax Tree) -> Seit PHP 7

Geparsed über PHPParser
(by Nikita Popov / nikic)

```
return $this->foo && self::bar();``
```

```
PhpParser\Node\Stmt\Return_  
└─ expr: PhpParser\Node\Expr\BinaryOp\BooleanAnd  
    └─ left: PhpParser\Node\Expr\PropertyFetch  
        └─ var: PhpParser\Node\Expr\Variable  
            └─ name: PhpParser\Node\Identifier  
    └─ right: PhpParser\Node\Expr\StaticCall  
        └─ class: PhpParser\Node\Name  
            └─ name: PhpParser\Node\Identifier  
        └─ args: array()
```

INSTANT UPGRADE TOOLS

RectorPHP

(by Tomas Votruba)

```
class Foo extends \Controller
{
    public function bar()
    {
        $GLOBALS['TL_DCA']['tl_content']
        ['palettes']['foo'] = '{foo_legend},foo;
        {bar_legend:hide},bar;{baz_Iegend:hide},baz';
    }
}
```

```

PhpParser\Node\Stmt\Class_(
  attrGroups: []
  flags: 0
  name: PhpParser\Node\Identifier( name: "Foo" )
  extends: PhpParser\Node\Name\FullyQualified( parts: ["Controller"] )
  implements: []
  stmts: [
    0: PhpParser\Node\Stmt\ClassMethod(
      attrGroups: []
      flags: MODIFIER_PUBLIC (1)
      byRef: false
      name: PhpParser\Node\Identifier( name: "bar" )
      params: []
      returnType: null
      stmts: [
        0: PhpParser\Node\Stmt\Expression(
          expr: PhpParser\Node\Expr\Assign(
            var: PhpParser\Node\Expr\ArrayDimFetch(
              var: PhpParser\Node\Expr\ArrayDimFetch(
                var: PhpParser\Node\Expr\ArrayDimFetch(
                  var: PhpParser\Node\Expr\ArrayDimFetch(
                    var: PhpParser\Node\Expr\Variable( name: "GLOBALS" )
                    dim: PhpParser\Node\Scalar\String_( value: "TL_DCA" )
                  )
                  dim: PhpParser\Node\Scalar\String_( value: "tl_content" )
                )
                dim: PhpParser\Node\Scalar\String_( value: "palettes" )
              )
              dim: PhpParser\Node\Scalar\String_( value: "foo" )
            )
            expr: PhpParser\Node\Scalar\String_( value: "{foo_legend},foo;{bar_legend:hide},bar;{baz_legend:hide},baz" )
          )
        )
      ]
    )
  ]
)

```

Beispiel

Rector Rules for Contao Open Source CMS

This project contains [Rector rules](#) for [Contao Open Source CMS](#) upgrades.

!! WARNING !! this is currently experimental, use at your own risk

Install

Install contao-rector via composer to your project:

```
composer require contao/contao-rector --dev
```



Available sets

Sets	Description
<code>ContaoSetList::CONTAO_49</code>	updates your code to compatibility with Contao 4.9
<code>ContaoSetList::CONTAO_413</code>	updates your code to compatibility with Contao 4.13
<code>ContaoSetList::CONTAO_50</code>	updates your code to compatibility with Contao 5.0
<code>ContaoSetList::ANNOTATIONS_TO_ATTRIBUTES</code>	converts Contao annotations (e.g. <code>@Hook("...")</code>) to attributes (e.g. <code>#[AsHook('...')]</code>)
<code>ContaoSetList::FQCN</code>	upgrades class namespaces from global (e.g. <code>\StringUtil</code>) to Contao (e.g. <code>\Contao\StringUtil</code>)

Available level sets

Level sets combine multiple changes for a specific Contao version, including dependencies like PHP, Symfony and Doctrine.

As an example, the `ContaoLevelSetList::UP_TO_CONTAO_413` will upgrade your code to PHP 7.4 and Symfony 5.4, since Contao 4.13 does not support any lower versions.

Available rules

Beispiel

rector.php

```
return RectorConfig::configure()
    ->withPhpSets(
        php83: true,
    )
    ->withAttributesSets(
        symfony: true,
        doctrine: true,
    )
    ->withSets([
        ContaoLevelSetList::UP_TO_CONTAO_53,
        ContaoSetList::ANNOTATIONS_TO_ATTRIBUTES,
    ])
    ->withPaths([
        __DIR__ . '/contao',
        __DIR__ . '/src',
    ])
    ->withImportNames(removeUnusedImports: true)
    ->withParallel()
;
```

Rector Regeln schreiben (optional)

Note to yourself, open the IDE

Fragerunde

Danke fürs Zuhören!

Speaker notes