

Formulare im Backend

Fabian Blechschmidt, IKONOSHIRT,
Magento Certified Developer
Magento-Stammtisch Berlin,
20. März 2012

The screenshot shows the Magento 0.8.17240 backend interface for editing a product. The top navigation bar includes 'Customers', 'Promotions', 'Newsletter', 'CMS', 'Reports', and 'System'. The product title is 'Zolof The Rock And Roll Destroyer: LOL Cat T-shirt (T Shirts)'. Action buttons include 'Back', 'Reset', 'Delete', 'Duplicate', and 'Save'. The 'General' tab is active, showing various product attributes:

Attribute	Value	Scope
Name *	Zolof The Rock And Roll Destroyer: LOL Cat T-shi	[STORE VIEW]
Model *	Zolof The Rock And Roll Destroyer: LOL Cat T-shi	[GLOBAL]
SKU *	zol_r_lrg	[GLOBAL]
Weight *	0.4400	[GLOBAL]
Status *	Enabled	[STORE VIEW]
Tax Class *	Taxable Goods	[STORE VIEW]
URL key	zolof-the-rock-and-roll-destroyer-lol-cat-t-shirt	[GLOBAL]
Visibility *	Nowhere	[STORE VIEW]
Allow Gift Message	Use config	[GLOBAL]
Manufacturer		[GLOBAL]
Set Product as New from Date		[GLOBAL]
Set Product as New to Date		[GLOBAL]

The footer of the page displays 'Magento ver. 0.8.17240' and a link to 'Connect with the'.

Formulare im Backend

Fabian Blechschmidt, IKONOSHIRT,
Magento Certified Developer
Magento-Stammtisch Berlin,
20. März 2012

The screenshot shows the Magento 0.8.17240 backend interface for editing a product. The top navigation bar includes 'Customers', 'Promotions', 'Newsletter', 'CMS', 'Reports', and 'System'. The product title is 'Zolof The Rock And Roll Destroyer: LOL Cat T-shirt (T Shirts)'. The 'General' tab is active, showing various product attributes. A red circle highlights the version number '0.8.17240' in the footer.

Attribute	Value	Scope
Name *	Zolof The Rock And Roll Destroyer: LOL Cat T-shi	[STORE VIEW]
Model *	Zolof The Rock And Roll Destroyer: LOL Cat T-shi	[GLOBAL]
SKU *	zol_r_lrg	[GLOBAL]
Weight *	0.4400	[GLOBAL]
Status *	Enabled	[STORE VIEW]
Tax Class *	Taxable Goods	[STORE VIEW]
URL key	zolof-the-rock-and-roll-destroyer-lol-cat-t-shirt	[GLOBAL]
Visibility *	Nowhere	[STORE VIEW]
Allow Gift Message	Use config	[GLOBAL]
Manufacturer		[GLOBAL]
Set Product as New from Date		[GLOBAL]
Set Product as New to Date		[GLOBAL]

Inhalt

- Wer bin ich? Der Erzähler.
- Was wollen wir? Das Ziel.
- Wie schaffen wir das? Die Geschichte.
- Beispiel: Folienfarben speichern

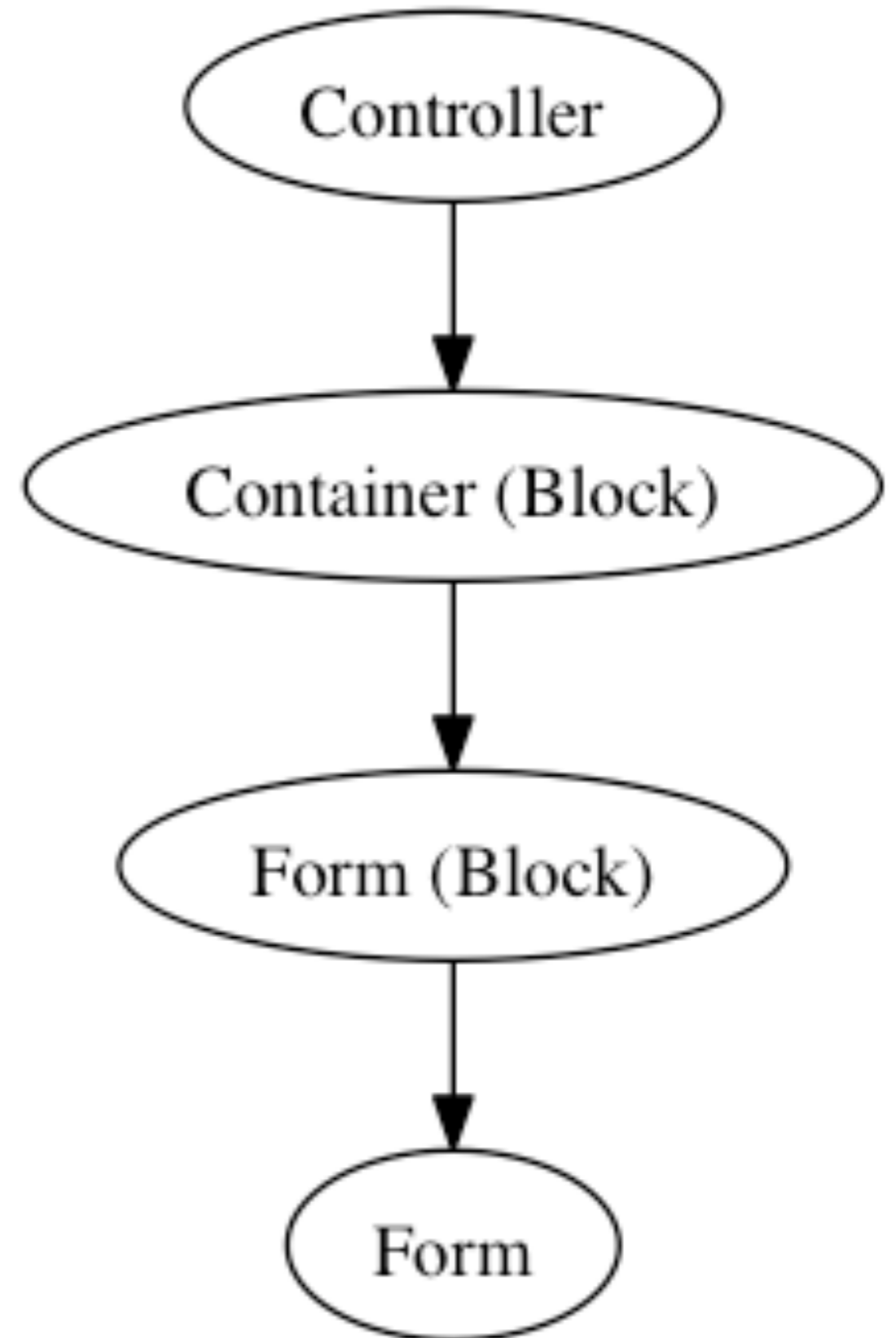
Der Erzähler

- Fabian Blechschmidt
- PHP seit 2004
- Freelancer seit 2008
- Magento seit Mitte 2011
- Certified Magento Developer (Okt. 2011)
- spielt gerne, aktuell mit
 - Magento und Symfony2



Die Geschichte

- Controller lädt Layout
- Layout lädt Container (Block)
- Container lädt Form (Block)
- Form erstellt Formular



Controller

- Controller kümmert sich um globale Verarbeitung
 - kann z.B. Laden von Model nach ID sein
 - oder auch einfach gar nichts:

```
5 | public function indexAction() {  
6 |     $this->loadLayout();  
7 |  
8 |     // HERE MORE  
9 |  
10 |     $this->renderLayout();  
11 | }
```

Blöcke - der Container

- alle Funktionsblöcke im Backend haben einen Container
- für Formulare:
 - X extends Mage_Adminhtml_Block_Widget_Form_Container
 - darin die Form:
 - Y extends Mage_Adminhtml_Block_Widget_Form

Widget_Form_Container

- „Administrationsschicht“
- Standard-Template: widget/form/container.phtml
- Standard-Buttons: Zurück, Reset, Löschen, Speichern
- Erzeugt die Widget_Form allein!
 - Sucht automatisch eine Form innerhalb des „Namespaces“
 - `$this->_blockGroup . '/' . $this->_controller . '_' . $this->_mode . '_form'`
 - `addCustomOption/adminhtml_foilcolor_edit_form`

Widget_Form

- „Arbeitsschicht“
- Erzeugt schnell und einfach gleiche Formulare
- Wrapper für Formulare und Fieldsets

Tipps für Container und Form

- Allgemeine Tipps:
 - VORSICHT mit Groß- und Kleinschreibung :-)
- Container (um Form zu erzeugen):
 - `$this->_objectId = 'foilcolor_id';`
 - `$this->_controller = 'adminhtml_foilcolor';`
 - `$this->_blockGroup = 'addCustomOption';`
- Form:
 - `$form->setValues($model->getData());`



Super Sache, gestern gelernt

Beispiel:

Folienfarben speichern

- T-Shirt wird mit Folie bedruckt
- n Farben zur Auswahl
- Verwaltung im Backend



✔ Your foilcolor has been updated successfully.

New Foilcolor

◀ Back

Reset

✔ Save Foilcolor

General Information

Foilcolor *

Color (#hex) *

Beispiel:
So soll's aussehen

Widget_Form_Container

```
1 <?php
2
3 class IkonoShirt_AddCustomOption_Block_Adminhtml_Foilcolor_Edit extends Mage_Adminhtml_Block_Widget_Form_Container {
4
5     public function __construct() {
6         parent::__construct();
7         $this->_objectId = 'foilcolor_id';
8         $this->_controller = 'adminhtml_foilcolor';
9         $this->_blockGroup = 'addCustomOption';
10
11
12
13         $this->_updateButton('save', 'label', Mage::helper('cms')->__('Save Foilcolor'));
14         //$this->_updateButton('delete', 'label', Mage::helper('cms')->__('Delete Foilcolor'));
15
16     }
17
18     /**
19      * Get edit form container header text
20      *
21      * @return string
22      */
23     public function getHeaderText() {
24         //if (Mage::registry('foilcolor_block')->getId()) {
25         //    return Mage::helper('addCustomOption')->__("Edit Foilcolor");
26         //} else {
27         return Mage::helper('addCustomOption')->__('New Foilcolor');
28         //}
29     }
30
31 }
32
```

Widget_Form_Container

- Beispiel in PHPStorm

Man beachte: Layout-XML OHNE Form_Widget!

- `<?xml version="1.0"?>`
- `<layout>`
- `<addcustomoption_adminhtml_foilcolor_index>`
- `<update handle="addcustom_foilcolor_index"/>`
- `<reference name="content">`
- `<block type="addCustomOption/adminhtml_foilcolor_edit" name="foilcolor_edit">`
- `</block>`
- `</reference>`
- `</addcustomoption_adminhtml_foilcolor_index>`
- `</layout>`

Widget_Form

```
1 <?php
2
3 class IkonoShirt_AddCustomOption_Block_Adminhtml_Foilcolor_Edit_Form extends Mage_Adminhtml_Block_Widget_Form {
4
5     /**
6      * Init form
7      */
8     public function __construct() {
9         parent::__construct();
10        $this->setId('foilcolor_form');
11        $this->setTitle(Mage::helper('cms')->__('Foilcolor Information'));
12    }
13
14    protected function _prepareForm() {
15        $model = Mage::getSingleton('addCustomOption/foilcolor')->load(1);
16        $form = new Varien_Data_Form(array(
17            'id' => 'edit_form', 'action' => $this->getData('action'), 'method' => 'post'
18        ));
19        $form->setHtmlIdPrefix('foilcolor_');
20
21        $fieldset = $form->addFieldset('base_fieldset', array(
22            'legend' => Mage::helper('addCustomOption')
23                ->__('General Information'),
24        ));
25
26        if ($model->getFoilcolorId()) {
27            $fieldset->addField('foilcolor_id', 'hidden', array(
28                'name' => 'foilcolor_id',
29            ));
30        }
31
32        $fieldset->addField('name', 'text', array(
33            'name' => 'name',
34            'label' => Mage::helper('cms')->__('Foilcolor'),
35            'title' => Mage::helper('cms')->__('Foilcolor'),
36            'required' => true,
37        ));
38
39        $fieldset->addField('color', 'text', array(
40            'name' => 'color',
41            'label' => Mage::helper('cms')->__('Color (#hex)'),
42            'title' => Mage::helper('cms')->__('Color'),
43            'required' => true,
44        ));
45
46        $form->setValues($model->getData());
47        $form->setUseContainer(true);
48        $this->setForm($form);
49        return parent::_prepareForm();
50    }
51 }
52 }
```


Widget_Form

- Beispiel in PHPStorm

Alles auf Anfang! - Controller

- HTTP-Methode prüfen
- Daten prüfen
- Model laden
- Werte setzen
- Model speichern
- Erfolgs/Fehlermeldung speichern
- **Weiterleiten!!**

HTTP

- POST-Request
 - Verarbeiten der Daten
- Weiterleitung 301
 - Wechsel auf GET
 - Refresh problemfrei möglich!



Bestätigen

Um diese Seite anzuzeigen, müssen die von Firefox gesendeten Daten erneut gesendet werden, wodurch alle zuvor durchgeführten Aktionen wiederholt werden (wie eine Suche oder eine Bestellaufgabe).

Abbrechen

Erneut senden

```

12
13 public function saveAction() {
14     $post = $this->getRequest()->getPost();
15     try {
16         if (empty($post)) {
17             Mage::throwException($this->__('Invalid form data.'));
18         }
19
20         $model = Mage::getModel('addCustomOption/foilcolor')->load($this->getRequest()->getParam('foilcolor_id'));
21         $model->setData($post);
22         $model->save();
23
24         $message = $this->__('Your foilcolor has been updated successfully. ');
25         Mage::getSingleton('adminhtml/session')->addSuccess($message);
26     } catch (Exception $e) {
27         Mage::getSingleton('adminhtml/session')->addError($e->getMessage());
28     }
29     $this->_redirect('*/*');
30 }

```

Daten prüfen, Model laden, Werte setzen, Model speichern,
Erfolgsmeldung ausgeben, Weiterleiten



Fragen? Antworten!
Danke für die Aufmerksamkeit.

Quellen und weitere Ressourcen

- [http://www.magentocommerce.com/wiki/5 - modules and development/0 - module development in magento/how to create an admin form module](http://www.magentocommerce.com/wiki/5_-_modules_and_development/0_-_module_development_in_magento/how_to_create_an_admin_form_module)
- <http://codemagento.com/2011/02/grids-and-forms-in-the-admin-panel/>
- <http://www.webguys.de/magento/turchen-13-feldtypen-und-formularen-im-magento-backend/>
- <http://blog.ebene7.com/2011/03/03/magento-datumsfeld-mit-kalender-button-im-admin-formular/>
- <http://aui.de/code-snippets/ein-magento-datentyp-timeline-108-article>