TYPO3.Fluid - Feature # 37506

Author: Carsten Bleicker Category: ViewHelpers Created: 2012-05-26 Assigned To: Bastian Waidelich Updated: 2012-06-25 Due date:	Status:	Closed	Priority:	Won't have this time		
Created: 2012-05-26 Assigned To: Bastian Waidelich Due date: Has patch: No Subject: Building up Nested Forms with this ViewHelper f.e. you sould be able to pass one object to <f.form object="(person) name=" person"=""> and inside this form you can display subproperties 1.e. personName by passing and use general forms with single property namespace without . because of traversing. it is possible to use it in unlimited depth by: <f.form action="update" object="(person.name.supproperty)"> <vh.mestedform (person.name.supproperty)"="" name="aerosnName' object="> <vh.mestedform (person.name.supproperty)"="" name="aerosnName' object="> <vh.mestedform (person.name.supproperty)"="" name="aerosnName' object="> <vh.mestedform (person.name.supproperty)"="" name="aerosnName' object="> */threstedForm name="aerosnName' object="(person.name.supproperty)"> <vh.mestedform (person.name.supproperty]"="" name="aerosnName' object="> ####################################</vh.mestedform></vh.mestedform></vh.mestedform></vh.mestedform></vh.mestedform></f.form></f.form>	Author:			ViewHelpers		
Updated: 2012-06-25 Due date: Has patch:: No Subject:: Building up Nested Forms with this ViewHelper Le, you sould be able to pass one object to -f:form object="[person] name="person"> and inside this form you can display subproperties 1.e. personName by passing and uside this form you can display subproperties 1.e. personName by passing and use general forms with single property namespace withoud . because of traversing. it is possible to use it in unlimited depth by: <fform action="update" object="(person.name.supproperty]"> <</fform>		2012-05-26				
Has patch: No Stubject: Building up Nested Forms with this ViewHelper Description Just want to know what you think about this viewhelper 1e. you sould be able to pass one object to <f:form object="[person] name=" person"=""> and inside this form you can display subproperties f.e. personName by passing and use general forms with single property namespace withoud . because of traversing. it is possible to use it in unlimited depth by: <f.form action="update" object="[person.name]"> vih.nestedForm name="personName" object="[person.name]"> vih.nestedForm name="anothersubproperty" object="[person.name].supproperty]"> .therestedForm name="anothersubproperty" object="[person.name]"> .vih.nestedForm name="anothersubproperty" object="[person.name]"> .vih.nestedForm name="anothersubproperty" object="[person.name]"> .therestedForm name="personName" object="[person.name]"> &thythnestedForm name="personName" object="[person.name]"> &tifieldset> &tifieldset> &tifieldset> &tifieldset> &tifieldset> &</f.form></f:form>	Updated:	2012-06-25	-			
Subject: Building up Nested Forms with this ViewHelper Description Just want to know what you think about this viewhelper f.e. you sould be able to pass one object to <f.form object="(person) name=" person"=""> and inside this form you can display subproperties 1.e. personName by passing and use general forms with single property ramespace without . because of traversing. it is possible to use it in unlimited depth by: <f.form (person)"="" action="update' object=" objectname="person"> <vh.nestedform name="subproperty" object="(person.name)"> <vh.nestedform name="anothersubproperty" object="(person.name)"> <vh.nestedform name="anothersubproperty" object="(person.name)"> <vh.nestedform name="subproperty" object="(person.name)"> <vh.nestedform name="subproperty" object="(person.name)"> <vh.nestedform name="subproperty" object="(person.name)"> <vh.nestedform name="subproperty" object="(person.name)"> */inestedForm name="personName" object="(person.name)"> &titylinestedForm name="personName" object="(person.name)"Sgtttattattattattattattattattattattattatt</vh.nestedform></vh.nestedform></vh.nestedform></vh.nestedform></vh.nestedform></vh.nestedform></vh.nestedform></f.form></f.form>	•	No				
Just want to know what you think about this viewhelper f.e. you sould be able to pass one object to <f.form object="(person) name=" person"=""> and inside this form you can display subproperties f.e. personName by passing and use general forms with single property namespace withoud . because of traversing. it is possible to use it in unlimited depth by: <f.form action="update" object="(person)" objectname="person"> </f.form></f.form>	-	Building up Nested Forms w	ith this ViewHelper			
<pre>te. you sould be able to pass one object to <f.form object="[person] name=" person"=""> and inside this form you can display subproperties f.e. personName by passing and use general forms with single property namespace withoud . because of traversing. it is possible to use it in unlimited depth by: </f.form></pre> <pre> f.e. personName by passing and use general forms with single property namespace withoud . because of traversing. it is possible to use it in unlimited depth by: </pre> <	Description					
<pre>te. you sould be able to pass one object to <f.form object="[person] name=" person"=""> and inside this form you can display subproperties f.e. personName by passing and use general forms with single property namespace withoud . because of traversing. it is possible to use it in unlimited depth by: </f.form></pre> <pre> f.e. personName by passing and use general forms with single property namespace withoud . because of traversing. it is possible to use it in unlimited depth by: </pre> <	lust want to know w	that you think about this viewboln	or			
and inside this form you can display subproperties f.e. personName by passing and use general forms with single property namespace withoud . because of traversing. It is possible to use it in unlimited depth by: Example usage: ####################################		•				
and use general forms with single property namespace withoud . because of traversing. It is possible to use it in unlimited depth by: Example usage: ####################################	-					
<pre>it is possible to use it in unlimited depth by: </pre> Example usage: #################################						
<pre> Example usage: ####################################</pre>	-		ce withoud . because of traversing.			
<vh:nestedform name="personName" object="{person.name}"> <vh:nestedform name="subproperty" object="{person.name.supproperty}"> <vh:nestedform name="anothersubproperty" object="{person.name.supproperty.anothersubproperty}"> Example usage: ####################################</vh:nestedform></vh:nestedform></vh:nestedform>	It is possible to use	It in unlimited depth by:				
<pre><vh:nestedform name="subproperty" object="{person.name.supproperty}"></vh:nestedform></pre>	<f:form action="upda</td><td>ate" object="{person}" objectname<="" td=""><td>e="person" ></td><td></td></f:form>	e="person" >				
<vh:nestedform name="anothersubproperty" object="{person.name.supproperty.anothersubproperty}"> Example usage: ####################################</vh:nestedform>						
Example usage: ####################################						
<pre>####################################</pre>	<vh:nestedf< td=""><td>orm name="anothersubproperty"</td><td>object="{person.name.supproperty</td><td>.anothersubproperty}"></td></vh:nestedf<>	orm name="anothersubproperty"	object="{person.name.supproperty	.anothersubproperty}">		
<pre>####################################</pre>	Example usage:					
1. Form ####################################			<i></i>			
<f:form action="update" object="{person}" objectname="person"> <vh:nestedform name="personName" object="{person.name}"> <fieldset> <fieldset> <fieldset> <fieldset> <:/vh:nestedForm> </fieldset></fieldset></fieldset></fieldset></vh:nestedform></f:form> ####################################				****		
<pre><vh:nestedForm name="personName" object="{person.name}"> <fieldset> <legend><f:translate id="person.name"></f:translate></legend> <fieldset> </fieldset> ####################################</fieldset></pre>	1.10111	*********	"""""""""""""""""""""""""""""""""""""""	******		
<pre><fieldset> <fieldset> <f:render partial="Model/Fieldset/PersonName/Default"></f:render> </fieldset> ####################################</pre>	<f:form action="upda</td><td>ate" object="{person}" objectname<="" td=""><td>e="person" ></td><td></td></f:form>	e="person" >				
<pre><legend><f:translate id="person.name" /></legend> <f:render partial="Model/Fieldset/PersonName/Default"></f:render> </vh:nestedForm></pre> 2. If uses in nestedForm it it reusable anywhere ###################################	<vh:nestedfor< td=""><td>m name="personName" object="{</td><td>{person.name}"></td><td></td></vh:nestedfor<>	m name="personName" object="{	{person.name}">			
<pre><f:render partial="Model/Fieldset/PersonName/Default" /> ####################################</pre>	<fieldset></fieldset>					
 ####################################	<legen< td=""><td>d><f:translate id="person.na</td><td>me"></f:translate></td><td></td></legen<>	d> <f:translate id="person.na</td><td>me"></f:translate>				
 ####################################	<f:renc< td=""><td>ler partial="Model/Fieldset/Person</td><td>Name/Default" /></td><td></td></f:renc<>	ler partial="Model/Fieldset/Person	Name/Default" />			
####################################	<td>></td> <td></td> <td></td>	>				
<pre>####################################</pre>	<td>rm></td> <td></td> <td></td>	rm>				
<pre>####################################</pre>						
1. Parial: Model/Fieldset/PersonName/Default # 2. If uses in nestedForm it it reusable anywhere ###################################		***	#######			
<label class="control-label" for="firstName"><f:translate id="personName.firstName"></f:translate>:</label> <!-- No need for person.name.firstName because namespace will be set by nestedForm-->	1. Parial: Model/F	Fieldset/PersonName/Default #				
No need for person.name.firstName because namespace will be set by nestedForm			#######################################	*************		
	<label class="contro</td><td>ol-label" for="firstName"><f:transla< td=""><td>ate id="personName.firstName" />:<</td><td></td></f:transla<></label>	ate id="personName.firstName" />:<				
				1>		

History

#1 - 2012-05-26 16:11 - Carsten Bleicker

- File NestedFormViewHelper.php added

Sorry for these stupid inline comments.

Usage now in the comments of attached php file.

#2 - 2012-05-26 16:14 - Carsten Bleicker

- Assigned To set to Bastian Waidelich

#3 - 2012-05-26 16:59 - Bastian Waidelich

- Assigned To deleted (Bastian Waidelich)

Hi Carsten,

why isn't that working for you:

1<f:form action='update' object="{person}" objectName="person" >
2 <f:form.textfield property="name.subproperty" />
3</f:form>

?

```
BTW: Please don't assign issues to us ;)
```

#4 - 2012-05-26 16:59 - Bastian Waidelich

- Status changed from New to Needs Feedback

#5 - 2012-05-26 17:08 - Carsten Bleicker

For example i have an address model and want to use this in an order model. deliveryAddress billingAddress

so i can build up just only one partial wich holds any fields. namespacing is set by nested viewhelper by adding the names from parent namespace.

so i can reduce my partialtemplates from 2 to only one partial.

call it "something like inlineEditing by reusing generalFieldset"

Bastian Waidelich wrote:

```
Hi Carsten,
why isn't that working for you:
[...]
?
```

BTW: Please don't assign issues to us ;)

#6 - 2012-05-26 17:27 - Bastian Waidelich

- Status changed from Needs Feedback to New

Carsten Bleicker wrote:

so i can build up just only one partial wich holds any fields. namespacing is set by nested viewhelper by adding the names from parent namespace.

I get you. But you should be able to achieve the same with some clever partials:

1<f:form action='update' object="{person}" objectName="person" >
2 <f:render partial="AddressForm" section="formFields" arguments="{propertyPrefix: 'person.'}" />
3</f:form>

Lets see what others think about this

#7 - 2012-05-26 17:31 - Carsten Bleicker

Damn :) Touché! i will try this. sometimes life could be so easy :P but maybe the viewhelper is still usefull if someone doesnt want to handle these property prefixes :)

Bastian Waidelich wrote:

Carsten Bleicker wrote:

so i can build up just only one partial wich holds any fields. namespacing is set by nested viewhelper by adding the names from parent namespace.

I get you. But you should be able to achieve the same with some clever partials: [...]

Lets see what others think about this

#8 - 2012-06-25 15:17 - Bastian Waidelich

- Status changed from New to Closed

- Assigned To set to Bastian Waidelich
- Priority changed from Could have to Won't have this time

Carsten Bleicker wrote:

Damn :)

Touché! i will try this.

I hope you were successful ;)

but maybe the viewhelper is still usefull if someone

doesnt want to handle these property prefixes :)

Absolutely. Lets keep it here for people to find. But I close the issue for now.

Thanks for your input!

Files

NestedFormViewHelper.php	11.3 kB	2012-05-26	Carsten Bleicker
NestedFormViewHelper.php	12 kB	2012-05-26	Carsten Bleicker