

TYPO3.Flow - Bug # 37257

Status:	Resolved	Priority:	Could have
Author:	Carsten Bleicker	Category:	Property
Created:	2012-05-17	Assigned To:	Christian Müller
Updated:	2012-12-10	Due date:	
PHP Version:			
Has patch:	Yes		
Complexity:			
Affected Flow version: (any)			
Subject:	NULL on IntegerConverter		
Description			
<p>hi folks, f.e. i have a model wich holds a property "maxOrders". NULL represents "unlimited" 0 represents 0 1 represents 1 ...</p> <p>so i have an input field where the user should be able to empty it with the result that db field contains NULL (unlimited)</p> <p>so in my humble oppinion the integer converter should convert nothing to null. maybe in combination with a check of @ORM\Column(nullable=true)? maybe the combination with @ORM\Column(nullable=true) could be also in any other converter?</p> <p>kind regards carsten</p>			

History

#1 - 2012-05-18 14:08 - Christian Müller

Indeed there should be a difference between "no entry made" and 0. For your usecase I would probably go a pragmatic way and set maxOrders to the highest possible integer value instead of NULL, but still I think there should be a way to find out if someone entered nothing or 0.

#2 - 2012-05-18 14:20 - Carsten Bleicker

- File IntegerConverter.php added

Christian Mueller wrote:

Indeed there should be a difference between "no entry made" and 0. For your usecase I would probably go a pragmatic way and set maxOrders to the highest possible integer value instead of NULL, but still I think there should be a way to find out if someone entered nothing or 0.

To find out whats needed NULL, 0, 1, ... or what else i tried the attached TypeConverter ('NULL' is wrong in the attached Class. Just think there is NULL instead of 'NULL' :))

It returns NULL as expected but the Action Argument of persisted Elements does not recognize this NULL value and does not overwrite the existing value. NULL seems to be handeled as "This is NULL ... Drop it"

I tried to figure out where exactly this happens and i think it must be somewhere in MVC Mapping.
I digged deep but didnt find the position to give it a patch try :(

Kind regards
carsten

#3 - 2012-05-24 17:23 - Carsten Bleicker

- File *PropertyMapper.php* added

So just trying to figure out the general mapping of null if `@ORM\Column(nullable=true)`.
Do you agree with the comment in line LINE 172?

#4 - 2012-05-24 18:20 - Carsten Bleicker

- File *PropertyMapper.php* added
- Assigned To set to *Christian Müller*
- % Done changed from 0 to 50

Tried to patch it by reflectionService.
Is it okay to do it this way?

#5 - 2012-07-10 08:24 - Karsten Dambekalns

- Status changed from *New* to *Needs Feedback*
- Target version set to *2.0 beta 1*
- Has patch changed from *No* to *Yes*

#6 - 2012-12-10 18:44 - Christian Müller

- Tracker changed from *Feature* to *Bug*
- Status changed from *Needs Feedback* to *Resolved*
- Priority changed from *-- undefined --* to *Could have*
- % Done changed from 50 to 100
- Affected Flow version set to *(any)*

The behaviour for Integer and Float was fixed in `l178f616d0dd4acea90938384fb21600dd0f7c252`

I think the PropertyMapper shouldn't know about `ORM\Column` annotations, what if you don't use doctrine. Wrong behaviour is fixed, so I will close this ticket for now.

Files

IntegerConverter.php	1.7 kB	2012-05-18	Carsten Bleicker
PropertyMapper.php	13.9 kB	2012-05-24	Carsten Bleicker
PropertyMapper.php	14.1 kB	2012-05-24	Carsten Bleicker