TYPO3.Flow - Bug # 9093

Status:	Resolved	**Priority:**	Must have
Author:	Scott MacDonald	**Category:**	Persistence
Created:	2010-07-30	**Assigned To:**	Karsten Dambekalns
Updated:	2010-10-20	**Due date:**	

PHP Version:	
Has patch:	
Complexity:	
Affected Flow version:	

Subject:	UUID generated but object is null

Description

An object is created via a form and the create action redirects to another action to define the view; the second action receives a null argument.

After setting the second action to run even if it received null and running a debugger, I found that the UUID was generated and even properly appended to the URI, but the UUID refers to nothing.

[from newsgroup/mailing list]:

I saw an earlier thread that seemed to be similar to this but there were few details and it wasn't resolved.

I completed the flow3 1.0.0 alpha 7 tutorial on the web site then downloaded flow3 1.0.0 alpha 10 to work on my own project. In my project, I have a Disc object in the model:

```php
<?php
declare(ENCODING = 'utf-8');
namespace F3\TrackLibrary_PHP\Domain\Model;

/**
 * A disc
 *
 * @scope prototype
 * @entity
 */
class Disc {...
```

In DiscController.php, I have a create action and an edit action:

```php
    /**
     * Creates a new disc
     *
     * @param \F3\TrackLibrary_PHP\Domain\Model\Disc $disc A fresh Disc object which has not yet been added to the
repository
     * @return void
     */
    public function createAction(\F3\TrackLibrary_PHP\Domain\Model\Disc $disc) {
        $this->discRepository->add($disc);
        $this->disc = $disc;
        $this->redirect('edit', \NULL, \NULL, array('disc' => $disc));
    }
```

```
    /**
     * Edit a disc
     *
     * @param \F3\TrackLibrary_PHP\Domain\Model\Disc $disc
     * @return string
     */
    public function editAction(\F3\TrackLibrary_PHP\Domain\Model\Disc $disc) {...
```

The first two lines in createAction are things I put in there while debugging in order to confirm the issue. The code of editAction doesn't matter because I never get there. The problem is that the $disc argument passed to editAction is NULL. Running a debugger, I found that those first two lines of createAction don't stick: if I get rid of the argument for editAction in order to get it to execute, $this->disc is NULL and $this->discRepository is empty once I enter editAction.

Can anyone see what I'm doing wrong? I'm happy to provide more details, if needed.

Thanks,

Scott

Related issues:		
related to TYPO3.Flow - Task # 9457: Refactor type-conversion into ...Utility...	**Rejected**	**2010-08-25**

Associated revisions

Revision ebc416ef - 2010-08-30 16:37 - Karsten Dambekalns

[+BUGFIX] FLOW3 (MVC): Fix incorrect NULL value exception in Argument

When setValue() on Controller\Argument is used with an already mapped
object, an exception is thrown. This can happen since r929f4e34 added
recursive mapping of identities.

Change-Id: I86353d6aa6d3cbc8062f98ecffd10d669beabc4c
Resolves: #9093

History

#1 - 2010-08-12 18:00 - Karsten Dambekalns

- Status changed from New to Accepted

- Assigned To set to Karsten Dambekalns

- Start date deleted (2010-07-30)

#2 - 2010-08-16 12:59 - Karsten Dambekalns

Could you create a stripped down package that show the erratic behaviour and attach it here? That would be great.

#3 - 2010-08-16 13:00 - Karsten Dambekalns

- Status changed from Accepted to Needs Feedback

#4 - 2010-08-18 15:11 - Karsten Dambekalns

- Category set to Persistence

#5 - 2010-08-25 11:20 - Karsten Dambekalns

From the #FLOW3 IRC channel (edited for brevity):

> if you try the Blog example out using Alpha 11, I'm sure you'll get to see the problem
>
> the edit link seems to work well, but for some reason when the edit action in the controller is called, the object isn't setted properly...
>
> when I click on the link, it shows something like that : demo/post/edit?post[__identity]=2a3c7692-3450-47a6-a459-e75977bb3f48
>
> which is fine, it used to do that back then too
>
> but then it triggers that error : #1269616784: The value of argument "post" must be of type "F3\Demo\Domain\Model\Post", but was of type "NULL".
>
> if you change the @param to mixed instead of \F3\Demo\Domain\Model\Post, it's working
>
> according to the exception log, the type is F3\Demo\Domain\Model\Post_AOPProxy_Development

The proxy is a post, so it should work (it uses instanceof, after all)

#6 - 2010-08-25 12:38 - Karsten Dambekalns

The Argument->setValue() already receives the mapped object but cannot cope with this. Returning the value immediately if it is already instanceof $this->dataType solves the issue. But we then seem to miss the information we need for $this->origin - needs further investigation.

#7 - 2010-08-25 12:38 - Karsten Dambekalns

- Status changed from Needs Feedback to Accepted

- Target version set to 639

- Start date set to 2010-08-25

- Estimated time set to 2.00

#8 - 2010-08-25 12:39 - Karsten Dambekalns

- Priority changed from Should have to Must have

#9 - 2010-08-25 17:21 - Robert Lemke

- Target version changed from 639 to 1.0 alpha 12

#10 - 2010-08-30 16:45 - Karsten Dambekalns

- Status changed from Accepted to Resolved

- % Done changed from 0 to 100

Applied in changeset commit:"ebc416ef725248f4196ad85094fb5616f34c1beb".