

TYPO3.Flow - Feature # 37276

Status:	Closed	Priority:	Should have
Author:	Adrian Föder	Category:	Persistence
Created:	2012-05-18	Assigned To:	
Updated:	2012-05-21	Due date:	
PHP Version:			
Has patch:	No		
Complexity:			
Subject:	Customizable ValueObject hash		
Description			
<p>If I understand it correctly, the hash for a Value Object is calculated/done in Persistence\Aspect\PersistenceMagicAspect::generateValueHash().</p> <p>It seems (I'm just "thinking loudly"), that that hash is built from</p> <ul style="list-style-type: none">* the object's Persistence Identifier, if one* the concatenated representation of each constructor method argument, processed as following:<ul style="list-style-type: none">* a serialize(), if an array* its actual value, if not an object* its Persistence Identifier, if one* its timestamp, if DateTime <p>In my use case, a Doctrine Collection should also be part of the hash. I could imagine that this is not a seldom case in general, so I have two improvement suggestions:</p> <ul style="list-style-type: none">- directly support the Doctrine collection case with a sub loop, for example, or the Collection->toArray() method- or call a magic function, for example <pre>1/** 2 * @FLOW3\ValueObject 3 */ 4class FooBar { 5 public function __construct() { 6 // this is the constructor of this Value Object 7 \$this->that = 'some'; 8 } 9 10 /** 11 * Generates the value object hash. Arguments are as in constructor. 12 * @param mixed \$argument1 13 * @param mixed \$argument2 14 * @return string The value object hash 15 */ 16 public static function generateValueObjectHash() { 17 foreach (func_get_args()) ... whatever... 18 19 }</pre> <p>This public static method could be, if present, called by the Aspect; else the usual process is applied to retrieve the hash.</p>			

What do you think? Should a customizable VO hash be possible altogether?

Subtasks:

Feature # 37286: Include object properties in Value Object hash creation

Closed

History

#1 - 2012-05-18 16:41 - Christian Müller

biggest problem I would see is **awareness** that you can break your own data if you change or add that method...

#2 - 2012-05-18 16:52 - Adrian Förder

hm yeah agree.

With the current implementation, objects, if having no persistence identifier and not \DateTime, are disregarded however. For these cases there should at least spl_object_hash() be used; compare to Doctrine's ArrayCollection:

```
public function __toString()
{
    return __CLASS__ . '@' . spl_object_hash($this);
}
```

#3 - 2012-05-18 17:35 - Adrian Förder

- Status changed from New to Closed