

TYPO3.Flow - Feature # 47404

Status:	New	Priority:	Could have
Author:	Rafael Kähm	Category:	AOP
Created:	2013-04-20	Assigned To:	
Updated:	2013-05-21	Due date:	
PHP Version:			
Has patch:	No		
Complexity:	medium		
Subject:	Add getters and setters methods for introduced properties		
Description			
<p>Currently you must introduce methods over interface introduction and then wrap introduced methods with around advice.</p> <p>Like this:</p> <pre>1<?php 2namespace Acme\Test\Aspect; 3 4use TYPO3\Flow\Annotations as Flow; 5 6use TYPO3\Flow\Reflection\ObjectAccess; 7 8/** 9 * Inject attribute to every as "MyOwnTypeOfEntity" annotated entity 10 * 11 * @Flow\Scope("singleton") 12 * @Flow\Aspect 13 * @Flow\Introduce("Acme\Test\Aspect\PropertyIntroductionAspect->isMyOwnTypeOfEntity", interfaceName="Acme\Test\Aspect\PropertyIntroductionInterface") 14 */ 15class PropertyIntroductionAspect { 16 17 /** 18 * @Flow\Inject 19 * @var \TYPO3\Flow\Reflection\ReflectionService 20 */ 21 protected \$reflectionService; 22 23 /** 24 * @Flow\Pointcut("TYPO3\Flow\Persistence\Aspect\PersistenceMagicAspect->isEntityOrValueObject && classAnnotatedWith(Acme\Test\Annotations\MyOwnTypeOfEntity)") 25 */ 26 public function isMyOwnTypeOfEntity() {} 27 28 /** 29 * @var string 30 * @Flow\Introduce("Acme\Test\Aspect\PropertyIntroductionAspect->isMyOwnTypeOfEntity") 31 */ 32 protected \$introducedProp; 33 34 /** 35 * Around advice, implements the new method "newMethod" of the</pre>			

```

36 * "NewInterface" interface
37 *
38 * @param \TYPO3\Flow\AOP\JoinPointInterface $joinPoint The current join point
39 * @return void
40 * @Flow\Around("Acme\Test\Aspect\PropertyIntroductionAspect->isMyOwnTypeOfEntity &&
method(*->setIntroducedProp())")
41 */
42 public function setIntroducedPropImplementation(\TYPO3\Flow\AOP\JoinPointInterface $joinPoint) {
43     $introducedProp = $joinPoint->getMethodArgument('introducedProp');
44     $proxy = $joinPoint->getProxy();
45
46     ObjectAccess::setProperty($proxy, 'introducedProp', $introducedProp, TRUE);
47
48     $someResult = $joinPoint->getAdviceChain()->proceed($joinPoint);
49 }
50
51}

```

Acme\Test\Aspect\PropertyIntroductionInterface

```

1<?php
2namespace Acme\Test\Aspect;
3
4/**
5 * Inject community-attribute to every class
6 *
7 */
8interface PropertyIntroductionInterface {
9
10 /**
11 * @param string $introducedProp some string
12 * @return void
13 */
14 public function setIntroducedProp($introducedProp);
15
16}

```

then you can see following code in

(../Data/Temporary/Development/Cache/Code/Flow_Object_Classes/Acme_Test_Domain_Model_Model.php

```

1...
2 /**
3 * Autogenerated Proxy Method
4 */
5 public function setIntroducedProp($introducedProp) {
6
7     // FIXME this can be removed again once Doctrine is fixed (see
fixMethodsAndAdvicesArrayForDoctrineProxiesCode())
8     $this->Flow_Aop_Proxy_fixMethodsAndAdvicesArrayForDoctrineProxies();
9     if (isset($this->Flow_Aop_Proxy_methodIsInAdviceMode['setIntroducedProp'])) {
10     $result = NULL;
11
12     } else {

```

```

13     $this->Flow_Aop_Proxy_methodIsInAdviceMode['setIntroducedProp'] = TRUE;
14     try {
15
16         $methodArguments = array();
17
18         $methodArguments['introducedProp'] = $introducedProp;
19
20         $adviceChains = $this->Flow_Aop_Proxy_getAdviceChains('setIntroducedProp');
21         $adviceChain = $adviceChains['TYPO3\Flow\Aop\Advice\AroundAdvice'];
22         $adviceChain->rewind();
23         $joinPoint = new \TYPO3\Flow\Aop\JoinPoint($this, 'Acme\Test\Domain\Model\Model', 'setIntroducedProp',
$methodArguments, $adviceChain);
24         $result = $adviceChain->proceed($joinPoint);
25
26     } catch (\Exception $e) {
27         unset($this->Flow_Aop_Proxy_methodIsInAdviceMode['setIntroducedProp']);
28         throw $e;
29     }
30     unset($this->Flow_Aop_Proxy_methodIsInAdviceMode['setIntroducedProp']);
31 }
32 return $result;
33 }
34...

```

as you can see there are too many steps and too many operations to add a simple getter and setter methods.

my suggestion:

add options to *TYPO3\Flow\Annotations\Introduce* :

addGetterIntroduction or **addSetterIntroduction** for manually method introductions

skipGetterIntroduction or **skipSetterIntroduction** for automatically method introductions

History

#1 - 2013-05-21 13:28 - Robert Lemke

- Target version deleted (2.1)