

TYPO3 Core - Bug #67337

hasProperty and isProperty not working on 7.3?

2015-06-07 16:09 - Tobias Braumann

| | | | |
|------------------------|-------------------|------------------------|------------|
| Status: | Closed | Start date: | 2015-06-07 |
| Priority: | Must have | Due date: | |
| Assignee: | Mathias Schreiber | % Done: | 0% |
| Category: | Fluid | Estimated time: | 0.00 hour |
| Target version: | 7.3 (Packages) | Complexity: | medium |
| TYPO3 Version: | 7 | Is Regression: | No |
| PHP Version: | 5.6 | Sprint Focus: | |
| Tags: | | | |

Description

According to TYPO3 CMS 7.1 - What's New and to the Ticket [#56529](#) it should be possible to use hasProperty and isProperty inside Fluid without adding a get the object method. But i didn't get it to work.

Is it bug or was it remove again?

My code looks like this:

```
@
class Collector implements CollectorInterface {
....
public function hasConfiguration() {
return true;
}

public function isTest() {
return true;
}
....
}
@
I created an instance of Collector with the objectManager.
```

In fluid it looks like this:

```
<f:debug>{collector.isTest}</f:debug>
<f:debug>{collector.hasConfiguration}</f:debug>
```

always get null, with get at the beginning of the methods it still works fine

History

#1 - 2015-06-07 16:44 - Mathias Schreiber

- Status changed from New to Needs Feedback
- Assignee set to Mathias Schreiber

Shouldn't is be the other way around?

Check this: <https://review.typo3.org/#/c/28036/4/typo3/sysex/extbase/Classes/Reflection/ObjectAccess.php>

IMO the method can be public function test().
Or I get it all wrong

#2 - 2015-06-08 03:44 - Tobias Braumann

I think you got i wrong. The Presentation said:

Method has() in ObjectAccess

For the usage in Fluid, object.property and object.isProperty already support the following methods:

- isProperty() # is isn't working too

- getProperty() # only this works with object.property
New since TYPO3 CMS 7.1: hasProperty()
This calls method \$object->hasProperty() if object.hasProperty is used in Fluid

#3 - 2015-06-08 09:08 - Mathias Schreiber

from reading the code (not the presentation) I take these possibilities:

```
$getterMethodName = 'has' . ucfirst($propertyName);  
if (is_callable(array($subject, $getterMethodName))) {  
    return $subject->{$getterMethodName}();  
}
```

To me it seems that the propertyName is "test" and the respective getter is build to be "hasTest"

```
if (substr($methodName, 0, 3) === 'has') {  
    $declaredPropertyNames[] = lcfirst(substr($methodName, 3));  
}
```

The other way around.

You go "hasTest" and the code builds (**mind the substr**) "test" from that.

#4 - 2015-06-08 09:39 - Anja Leichsenring

You need a property declared to be returned by the method. So judging by the example you gave, I guess you missed the property declaration. Try this:

```
protected $property = TRUE;  
  
protected $anotherProperty = array();  
  
public function isProperty() {  
    return $this->property();  
}  
  
public function hasAnotherProperty() {  
    return $this->anotherProperty();  
}
```

#5 - 2015-06-08 10:39 - Anja Leichsenring

Finally I got the failure here. In fluid, you always request the property, the inner implementation is not known. You do <f:debug>{property}</f:debug> and this is it. The system will find out what kind of getter to call, by simply looking up the provided methods. This is not known and not needed in fluid.

#6 - 2015-06-08 10:45 - Anja Leichsenring

- Status changed from Needs Feedback to Closed

this is no bug, but user error. Don't call methods in fluid templates.

#7 - 2015-06-08 18:52 - Tobias Braumann

Thx, now i know what Mathias mean. You are both right! I tested it and it works, but it didn't feel right and i think this behavior differs from what is describe in [#56529](#) or what stands in the official TYPO3 CMS 7.1 - What's New presentation.