

TYPO3.Flow - Bug # 46008

Status:	Resolved	Priority:	Should have
Author:	Christopher Hlubek	Category:	Reflection
Created:	2013-03-04	Assigned To:	Sebastian Kurfuerst
Updated:	2013-07-16	Due date:	
PHP Version:			
Has patch:	No		
Complexity:			
Affected Flow version: Git master			
Subject:	Reflection should resolve relative namespaces		
Description			
<p>Annotations (e.g. @var) with relative class names are not resolved correctly under the namespace of the given class. To have a correct handling of namespaces in PHP we should support the usage of relative class names in these places.</p> <pre>1 namespace MyVendor\MyPackage\Domain\Model; 2 3 class Foo { 4 5 /** 6 * @var Bar 7 */ 8 protected \$bar; 9 10 }</pre> <pre>1 namespace MyVendor\MyPackage\Domain\Model; 2 3 class Bar { 4 5 }</pre>			
<p>Since this is a problem for some people (e.g. also PhpStorm uses relative class names when auto-completing) I consider this a bug we should solve in the framework.</p>			
Related issues:			
related to TYPO3.Flow - Bug # 50909: Dependency injection for imported namesp...		Resolved	2013-08-08
related to TYPO3.Flow - Bug # 57034: Dependency injection for imported namesp...		Resolved	2014-03-18
related to Core - Bug # 57856: @inject does not work relative to current name...		Under Review	2014-04-12

Associated revisions

Revision bc8f7164 - 2013-07-15 11:37 - Sebastian Kurfuerst

[FEATURE] Resolve relative namespaces and use statements in @var & @param annotations

Reflection should expand shortened or namespaced @var & @param annotations to their fully qualified class name.

This has a direct benefit in the following areas (all covered by tests):

- Reflection returns the fully qualified class name for @var and @param annotations

- Reflection returns the fully qualified property types for Domain Models inside the class schema
- Dependency Injection using @Flow\Inject works with shortened class names
- Property mapping works for shortened class names in type hint & @param annotation

Resolves: #46008

Releases: master

Change-Id: I28fd7c7993ea79714cc62032fcc332ac8cf9d4bc

Revision 751b4b55 - 2013-09-19 12:48 - Bastian Waidelich

[BUGFIX] Fix resolving of relative namespaces in parent classes

With I28fd7c7993ea79714cc62032fcc332ac8cf9d4bc the ReflectionService resolves relative and aliased namespaces automatically.

This doesn't work for inherited classes though because the code only considered the current class, not it's parents.

With this change code like this will work::

```

use TYPO3\Flow\Security\Context;
class SomeParentClass {
/**
- @var Context
- @Flow\Inject
*/
protected $securityContext;
// ...
}
class SomeChildClass extends SomeParentClass {
}

```

Change-Id: Ie4b931a03a94d506890e996b9d8bae7d0d1a2e14

Fixes: #50909

Related: #46008

Releases: master

Revision ca5dd2ca - 2014-03-18 13:11 - Bastian Waidelich

[BUGFIX] Fix resolving of imported collection types

This change extends the ReflectionService so that it is able to expand our custom notation for collection types::

```

CollectionType&lt;ElementType&gt;

```

Background:

In the Coding Guidelines we encourage developers to import namespaces in order to increase readability.

With #46008 the ReflectionService is able to resolve relative and imported namespace (with #50909 this also works for abstract classes).

But this does not work for our custom notation of Collection types yet.
With this fix relative and imported class names are supported also for
our custom notation of collection types and the above would be
expanded to::

```
\CollectionTypeNamespace\CollectionType<\ElementTypeNamespace\ElementType>
```

Change-Id: I4e228d6846776d2500d7e8aec2bc88df170bc9de

Fixes: #57034

Related: #46008

Related: #50909

Releases: master, 2.2, 2.1

Revision 0b45aa53 - 2014-03-20 11:00 - Bastian Waidelich

[BUGFIX] Fix resolving of imported collection types

This change extends the ReflectionService so that it is able to
expand our custom notation for collection types::

```
CollectionType&lt;ElementType&gt;
```

Background:

In the Coding Guidelines we encourage developers to import namespaces
in order to increase readability.

With #46008 the ReflectionService is able to resolve relative and
imported namespace (with #50909 this also works for abstract classes).

But this does not work for our custom notation of Collection types yet.
With this fix relative and imported class names are supported also for
our custom notation of collection types and the above would be
expanded to::

```
\CollectionTypeNamespace\CollectionType<\ElementTypeNamespace\ElementType>
```

Change-Id: I4e228d6846776d2500d7e8aec2bc88df170bc9de

Fixes: #57034

Related: #46008

Related: #50909

Releases: master, 2.2, 2.1

Revision 4935af46 - 2014-03-20 11:00 - Bastian Waidelich

[BUGFIX] Fix resolving of imported collection types

This change extends the ReflectionService so that it is able to
expand our custom notation for collection types::

```
CollectionType&lt;ElementType&gt;
```

Background:

In the Coding Guidelines we encourage developers to import namespaces

in order to increase readability.

With #46008 the ReflectionService is able to resolve relative and imported namespace (with #50909 this also works for abstract classes).

But this does not work for our custom notation of Collection types yet. With this fix relative and imported class names are supported also for our custom notation of collection types and the above would be expanded to::

```
\CollectionTypeNamespace\CollectionType<\ElementTypeNamespace\ElementType>
```

Change-Id: I4e228d6846776d2500d7e8aec2bc88df170bc9de

Fixes: #57034

Related: #46008

Related: #50909

Releases: master, 2.2, 2.1

History

#1 - 2013-04-04 10:44 - Bastian Waidelich

Probably we should also consider **use** statements:

```
1 namespace MyVendor\MyPackage\Domain\Model;
2
3 use MyVendor\MyPackage\Bar as Baz;
4
5 class Foo {
6
7     /**
8      * @var Baz
9      */
10    protected $baz;
11 }
```

but it seems PHP itself is buggy there <https://bugs.php.net/bug.php?id=63665>

#2 - 2013-04-04 10:55 - Bastian Waidelich

Bastian Waidelich wrote:

```
| but it seems PHP itself is buggy there https://bugs.php.net/bug.php?id=63665
```

This bug report seems is about something different it seems. This works as expected:

```
1 namespace Foo\Bar;
2
3 class Baz {
4
```

```
5}
6
7namespace MyPackage;
8use Foo\Bar\Baz as Quux;
9
10class Test {
11
12 /**
13  * @param Quux $param
14  * @return void
15  */
16 public function test(Quux $param) {
17 }
18}
19
20$reflectionMethod = new \ReflectionMethod('MyPackage\Test', 'test');
21echo current($reflectionMethod->getParameters())->getClass();
```

Result:

```
Class [ class Foo\Bar\Baz ] { @@ test.php 4-6 - Constants [0] {} - Static properties [0] {} - Static methods [0] {} - Properties [0] {} - Methods [0] {} }
```

#3 - 2013-04-25 11:43 - Sebastian Kurfuerst

- Status changed from New to Accepted
- Assigned To set to Sebastian Kurfuerst

I have a patch for this in the works; almost ready.

#4 - 2013-04-25 12:10 - Bastian Waidelich

Sebastian Kurfuerst wrote:

| I have a patch for this in the works; almost ready.

Great! Looking forward to this one

#5 - 2013-04-25 15:52 - Gerrit Code Review

- Status changed from Accepted to Under Review

Patch set 1 for branch **master** has been pushed to the review server.

It is available at <https://review.typo3.org/20158>

#6 - 2013-07-12 12:51 - Gerrit Code Review

Patch set 2 for branch **master** has been pushed to the review server.

It is available at <https://review.typo3.org/20158>

#7 - 2013-07-12 12:54 - Gerrit Code Review

Patch set 3 for branch **master** has been pushed to the review server.

It is available at <https://review.typo3.org/20158>

#8 - 2013-07-15 11:38 - Gerrit Code Review

Patch set 4 for branch **master** has been pushed to the review server.

It is available at <https://review.typo3.org/20158>

#9 - 2013-07-16 11:37 - Sebastian Kurfuerst

- *Status changed from Under Review to Resolved*

- *% Done changed from 0 to 100*

Applied in changeset commit:bc8f7164dea7331823b6170aa76833ffdc6c9658.