

TYPO3.Flow - Bug #39005

Proxies for Http\Request not used

2012-07-16 22:46 - Marco Falkenberg

Status:	Resolved	Start date:	2012-07-16
Priority:	Should have	Due date:	
Assignee:	Karsten Dambekalns	% Done:	100%
Category:	Http	Estimated time:	0.00 hour
Target version:	1.1	Complexity:	
PHP Version:			
Has patch:	No		

Description

The proxies for the Http\Request are generated but not used. So you are not able to use AOP for this class. For example try this:

```
/**
 * @param \TYPO3\FLOW3\Aop\JoinPointInterface $joinPoint The current joinpoint
 *
 * @FLOW3\Around("method(TYPO3\FLOW3\Http\Request->decodeBodyArguments())")
 * @return mixed Result of the advice chain
 */
public function transformRequestBody(\TYPO3\FLOW3\Aop\JoinPointInterface $joinPoint) {
    /** code **/
}
```

The advices are never called.

The bug is located in the Http\RequestHandler. There the Request is generated through a static constructor method.

Maybe the request should be instantiated by the object manager while resolving the dependencies like the other properties of the request handler.

Like this:

```
protected function resolveDependencies() {
    ...
    $this->request = $objectManager->get('TYPO3\FLOW3\Http\Request');
}
```

Then you could feed the constructor of the request with a kind of "RequestVariablesLocator" which encapsulates the superglobals. When it's a singleton it get's injected.

Like this:

```
public function __construct(\TYPO3\FLOW3\Mvc\RequestVariablesLocator $requestVariableLocator)
{
    $server = $requestVariableLocator->getServerVariables();

    $this->headers = Headers::createFromServer($server);
    $this->setMethod(isset($server['REQUEST_METHOD']) ? $server['REQUEST_METHOD'] : 'GET');
    $protocol = (isset($server['SSL_SESSION_ID']) || (isset($server['HTTPS']) && ($server['HTTPS'] === 'on' || strcmp($server['HTTPS'], '1') === 0))) ? 'https' : 'http';
    $this->uri = new Uri($protocol . '://' . (isset($server['HTTP_HOST']) ? $server['HTTP_HOST'] : 'localhost') . str_replace('/index.php', '', (isset($server['REQUEST_URI']) ? $server['REQUEST_URI'] : '/')));
    $this->server = $server;
    $this->arguments = $this->buildUnifiedArguments($requestVariableLocator->getGetVariables(), $requestVariableLocator->getPostVariables(), $requestVariableLocator->getPostVariables());
}
```

Related issues:

Related to TYPO3.Flow - Task #39121: Encapsulate PHP superglobals for Http\Re...

Rejected

2012-07-20

Associated revisions

Revision 868e073b - 2012-07-30 11:51 - Karsten Dambekalns

[BUGFIX] Disable proxy building for Http classes

This disables proxy building for (most of) the classes in the Http part of FLOW3.

They are not needing it and some would circumvent the built proxy because they are instantiated statically.

Change-Id: Ib632caaae6d20746760c2cdb1513f8b35dcd2626

Fixes: #39005

Releases: 1.1, 1.2

Revision aaa30271 - 2012-08-07 16:41 - Karsten Dambekalns

[BUGFIX] Disable proxy building for Http classes

This disables proxy building for (most of) the classes in the Http part of FLOW3.

They are not needing it and some would circumvent the built proxy because they are instantiated statically.

Change-Id: Ib632caaae6d20746760c2cdb1513f8b35dcd2626

Fixes: #39005

Releases: 1.1, 1.2

History

#1 - 2012-07-30 08:26 - Karsten Dambekalns

- Status changed from *New* to *Needs Feedback*

- Assignee changed from *Christian Müller* to *Robert Lemke*

Robert - I think HttpRequest could simply get a Proxy(false) annotation and we live with the fact that this makes AOP on this class impossible. IMHO acceptable.

#2 - 2012-07-30 09:22 - Robert Lemke

Karsten Dambekalns wrote:

Robert - I think HttpRequest could simply get a Proxy(false) annotation and we live with the fact that this makes AOP on this class impossible. IMHO acceptable.

yes, correct, just go ahead! I think the same goes for all other classes in the Http namespace.

#3 - 2012-07-30 11:19 - Karsten Dambekalns

- Status changed from *Needs Feedback* to *Accepted*

- Assignee changed from *Robert Lemke* to *Karsten Dambekalns*

- Target version set to *1.1*

#4 - 2012-07-30 11:51 - 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 <http://review.typo3.org/13344>

#5 - 2012-07-30 14:35 - Karsten Dambekalns

- Status changed from *Under Review* to *Resolved*

- % Done changed from *0* to *100*

Applied in changeset [868e073b9877d562c6ed65a5158c1c67b3a501a8](#).

#6 - 2012-08-07 16:41 - Gerrit Code Review

- Status changed from Resolved to Under Review

Patch set 1 for branch **FLOW3-1.1** has been pushed to the review server.
It is available at <http://review.typo3.org/13501>

#7 - 2012-08-14 11:35 - Karsten Dambekalns

- Status changed from Under Review to Resolved

Applied in changeset [aaa3027158ee86770f5c413812f20bf2afb28f47](#).