

TYPO3.Flow - Bug # 3121

Status:	Resolved	Priority:	Should have
Author:	Bastian Waidelich	Category:	MVC
Created:	2009-04-17	Assigned To:	
Updated:	2010-10-20	Due date:	
PHP Version:			
Has patch:			
Complexity:			
Affected Flow version:			
Subject:	canProcessRequest is not called in ActionControllers		
Description			
<p>It seems, canProcessRequest() is never called in Action Controllers, so CLI-Controllers can be called through the browser currently. In processRequest() of \F3\FLOW3\MVC\Controller\ActionController s.th. like</p> <pre>if (!\$this->canProcessRequest(\$request)) throw new \F3\FLOW3\MVC\Exception\UnsupportedRequestType(...)</pre> <p>is missing. Maybe it is possible to add a parent::processRequest(\$request, \$response) instead, as there is quite some duplicated code in both Classes.</p>			

Associated revisions

Revision 6fd4ce49 - 2009-04-22 17:03 - Robert Lemke

- FLOW3: Raised required PHP version to 5.3.0RC1
- FLOW3: (Error) The \F3\FLOW3\Error\Error class now supports __toString()
- FLOW3: (MVC) Finished implementation of validation rules in the action methods. Now these rules are actually enforced. Now really resolves #3092.
- FLOW3: (MVC) The ActionController now checks if it can process the request – didn't do that before. Fixes #3121.
- FLOW3: (Property) The property mapper's mapAndValidate() method now actually validates the source.
- FLOW3: (Reflection) The Reflection Service now detects class file changes again. (Problem was that the settings were not injected anymore because autowiring was turned off for the Reflection Service in the Objects.yaml) Fixes #3152.
- FLOW3: (Validation) Removed the Validation\Errors class - now a simple array is used instead.
- FLOW3: (Validation) Removed the \$errors argument from FilterInterface::filter(). More changes to this interface will follow.
- FLOW3: (Validation) Validation options must be set with the new setOptions() method instead of passing them to the isValid() method.
- FLOW3: (Validation) The errors array is not passed to the isValid() method of a validator anymore - instead it can be retrieved by a new getErrors() method which is defined in the ValidatorInterface.
- FLOW3: (Validation) Validators may now be singleton or prototype. Marked all those validators as prototype which require options and let others be singletons.
- FLOW3: (Validation) Removed the public resolveValidatorClassName() method from the ValidatorResolver class.
- FLOW3: (Validation) !!! changed the option names of the StringLength validator: now they are "minimum" and "maximum"
- FLOW3: (Validation) Changed the ObjectValidatorInterface: canValidateType() is now canValidate() and asks for an objects instead of a class name.
- FLOW3: (Validation) The first argument of the Validator Resolver's createMethodValidator() has changed: Now either the short name or a fully qualified object name of a *validator* is expected (instead of the data type)! Maybe also addresses #3130.
- FLOW3: (Documentation) Added a few lines of documentation and an example for the validation rules feature. More to come.
- Fluid: (Core) Fixed a test which was referring to the \F3\FLOW3\Validation\Errors class.

History

#1 - 2009-04-22 18:00 - Robert Lemke

- *Status changed from New to Resolved*
- *% Done changed from 0 to 100*

Applied in changeset r2175.