TYPO3.Flow - Feature #5660

Implement support for Object Converters

2009-12-07 19:01 - Robert Lemke

Status:	Resolved	**Start date:**	2009-12-07
Priority:	Must have	**Due date:**	
Assignee:	Robert Lemke	**% Done:**	100%
Category:	Property	**Estimated time:**	0.00 hour
Target version:	1.0 alpha 7		
PHP Version:		**Complexity:**	
Has patch:			

Description

Now finally the Property Mapper needs to support custom Object Converters (we previously called the Property Editors). We'll need them to convert a string to DateTime, a file upload array to a Resource object and probably a few more cases.

There should be an interface for Object Converters and all classes implementing this interface should automatically be registered as converters for the Property Mapper. Each converter can be responsible for one or more types (class names), but no two converters may be responsible for the same type.

Related issues:

Related to TYPO3.Flow - Feature #342: Support for file uploads	**Resolved**	**2008-02-20**
Related to TYPO3.Flow - Feature #27417: DateTime conversion support	**Resolved**	**2011-06-14**

Associated revisions

Revision 027a4016 - 2009-12-07 19:09 - Robert Lemke

[~TASK] FLOW3 (AOP): Removed the "Resource" sub package from the blacklisted sub packages because it now contains a class (Resource) which needs to be persistable.
[-FEATURE][!!!] FLOW3 (MVC): For now removed the request hash feature (HMAC) because it mocks a level of security for incoming data which it doesn't provide. The current mechanism effectively puts control over content security into Fluid templates and it doesn't belong there. Although there might be a need for a request hash, the content security must be implemented by other means. Relates to #4960 and relates to #5659.
[+FEATURE] FLOW3 (MVC): Implemented support for file uploads. Uploading files is cooperation between the Web Request Builder, the Property Mapper and the Resource sub package. The solution included in this commit provides handling of incoming files (including nested arguments) and transparent conversion into Resource objects. Resources (files) are only stored once, no matter how often they are uploaded or what original filename they carried. Still missing: view helper, documentation and automatic purging of unused resource files. Addresses #342.
[~API][!!!] FLOW3 (Property): Renamed the property mapper class to "PropertyMapper" (was just "Mapper" before). Relates to #5658
[+FEATURE] FLOW3 (Property): The Property Mapper now supports a mechanism called Object Converters. These convertes enable the mapper to convert strings, arrays or numbers to certain objects, for example a unix time stamp to a DateTime object. Resolves #5660.
[+FEATURE] FLOW3 (Reflection): Implemented the methods "isPropertySettable" and "isPropertyGettable" for the ObjectAccess class.
[~TASK] FLOW3 (Resource): Renamed the StreamWrapper class to StreamWrapperAdapter
[+FEATURE] FLOW3 (Resource): Implemented a ResourceObjectConverter which is capable of converting arrays or strings to Resource objects.
[+FEATURE] FLOW3 (Utility): Implemented a setValueByPath() method for the Array utilities class.
[+FEATURE] FLOW3 (Utility): Added support for the _FILES super global to the Environment class. The array of information about uploaded files can be obtained in a much cleaner way than PHP provides it by the new getUploadedFiles() method.

Revision 3536 - 2009-12-07 19:09 - Robert Lemke

[~TASK] FLOW3 (AOP): Removed the "Resource" sub package from the blacklisted sub packages because it now contains a class (Resource) which needs to be persistable.
[-FEATURE][!!!] FLOW3 (MVC): For now removed the request hash feature (HMAC) because it mocks a level of security for incoming data which it doesn't provide. The current mechanism effectively puts control over content security into Fluid templates and it doesn't belong there. Although there might be a need for a request hash, the content security must be implemented by other means. Relates to #4960 and relates to #5659.
[+FEATURE] FLOW3 (MVC): Implemented support for file uploads. Uploading files is cooperation between the Web Request Builder, the Property Mapper and the Resource sub package. The solution included in this commit provides handling of incoming files (including nested arguments) and transparent conversion into Resource objects. Resources (files) are only stored once, no matter how often they are uploaded or what original filename they carried. Still missing: view helper, documentation and automatic purging of unused resource files. Addresses #342.
[~API][!!!] FLOW3 (Property): Renamed the property mapper class to "PropertyMapper" (was just "Mapper" before). Relates to #5658
[+FEATURE] FLOW3 (Property): The Property Mapper now supports a mechanism called Object Converters. These convertes enable the mapper to convert strings, arrays or numbers to certain objects, for example a unix time stamp to a DateTime object. Resolves #5660.
[+FEATURE] FLOW3 (Reflection): Implemented the methods "isPropertySettable" and "isPropertyGettable" for the ObjectAccess class.
[~TASK] FLOW3 (Resource): Renamed the StreamWrapper class to StreamWrapperAdapter
[+FEATURE] FLOW3 (Resource): Implemented a ResourceObjectConverter which is capable of converting arrays or strings to Resource objects.
[+FEATURE] FLOW3 (Utility): Implemented a setValueByPath() method for the Array utilities class.
[+FEATURE] FLOW3 (Utility): Added support for the _FILES super global to the Environment class. The array of information about uploaded files can be obtained in a much cleaner way than PHP provides it by the new getUploadedFiles() method.

History

#1 - 2009-12-07 20:00 - Robert Lemke

- Status changed from Accepted to Resolved

- % Done changed from 0 to 100

Applied in changeset r3536.