

TYPO3.Flow - Feature # 35

Status:	Closed	Priority:	Should have
Author:	Robert Lemke	Category:	AOP
Created:	2007-03-01	Assigned To:	Robert Lemke
Updated:	2010-10-20	Due date:	
PHP Version:			
Has patch:			
Complexity:			
Subject:	Refactor T3P_Core_AOPFramework: Extract build methods		
Description			
The T3P_Core_AOPFramework got a bit too large by the time and it makes sense extracting the whole proxy code building methods into its own class. While doing that, the code snippets should also be externalized and put into template files where it makes sense.			

Associated revisions

Revision adde6ecc - 2009-01-15 15:24 - Robert Lemke

This is a big commit (sorry - not exactly Continuous Integration ...)

While all seems to be fine and stable, many features introduced in this commit are not complete. Especially AOP Proxy Caching, Logging (API), the System Logger and the Signal Slots mechanism need some further work.

- TYPO3 Distribution: Added a Log directory
- TYPO3 Distribution: Disabled security (for now) because some thing need to be refactored and tested again after this commit
- FLOW3: Further cleanup of the bootstrap. Addresses #2117
- FLOW3: Removed manual registration of objects from all classes and created an Objects.yaml configuration instead which declares the objects and configures their dependencies. This manual registration is necessary for objects such as the AOP Framework because at the time it is created, the Object Manager is not yet fully initialized.
- FLOW3: (AOP) Refactored and cleaned up the AOP initialization process. Resolves #441.
- FLOW3: (AOP) Completely rewrote all unit tests and added many new ones. Now no AOP test relies on the TestPackage anymore. Addresses #1868
- FLOW3: (AOP) Improved the automatic documentation rendering of the Proxy Class Builder. Resolves #35.
- FLOW3: (AOP) Externalized the creation of the AOP proxy and instead use the Caches.yaml configuration.
- FLOW3: (AOP) Removed all use of PHP's Reflection classes and replaced them with calls to the Reflection Service
- FLOW3: (AOP) Converted all classes involved in the proxy building process from static into dynamic calls
- FLOW3: (Cache) Introduced an (semi) automatic cache registration feature which assures that all necessary caches are registered when file changes are detected. In order to register a cache, just define it in the new "Caches.yaml" configuration file. See the FLOW3 package for examples. Resolves #1787
- FLOW3: (Cache) Wrote more tests for the File Backend. Resolves #1843
- FLOW3: (Configuration) Improved the performance of the special configuration management
- FLOW3: (Configuration) Added support for Caches.* configuration
- FLOW3: (Error) The exception handlers now log uncaught exceptions to the System Logger
- FLOW3: (Monitor) Introduced a new sub package "Monitor" which provides a mechanism for automatic detection of changes on monitored files. If monitored files or directories (the latter not yet implemented) change, a signal is emitted which can be caught by a slot taking further action. Currently all cache entries tagged with a class tag are flushed automatically if a class file changes.
- FLOW3: (Log) Introduced a special System Logger which is created automatically and can be injected if needed. Although it works quite well already, the general concept of injecting the logger and the LoggerInterface itself is not final! Watch for log files in the Data/Log/ directory when running TYPO3 / FLOW3 in development context.
- FLOW3: (Log) Introduced a Logger Factory for use in Objects configuration. Warning: the API might still change
- FLOW3: (Log) Made the SEVERITY constants conform to RFC3164 and PHP's LOG_* constants
- FLOW3: (Object) Refactored and detached more unit tests from the TestPackage. Addresses #2123

- FLOW3: (Object) Implemented an early configuration parse step which allows for configuring objects in the Objects.yaml file of the FLOW3 package even before the Package Manager is initialized.
- FLOW3: (Object) Introduced a lifecycle shutdown method (default is "shutdownObject") which can be used as a clean alternative to __destruct().
- FLOW3: (Reflection) Added new API methods: isClassReflected, getInterfaceNamesImplementedByClass, isMethodFinal, isMethodStatic, isMethodPublic, isMethodProtected, isMethodPrivate
- FLOW3: (Reflection) getMethods() now returns information about the method visibility
- FLOW3: (Security) Refactored the ACL Policy Service - might still need some final work
- FLOW3: (SignalSlot) Fixed some small issues in the Dispatcher and implemented support for System Logger
- FLOW3: (Utility) Fixed a bug in Arrays::getValueByPath
- Testing: Implemented a method buildAccessibleProxy() in the BaseTestCase which allows for conveniently calling protected methods of a class. See FLOW3 unit tests for examples.
- TestPackage: Removed many classes which became obsolete due to the AOP tests refactoring. refs #1868

History

#1 - 2007-03-01 15:24 - Robert Lemke

(In r159) * Core: Removed parsing support for "lifecycleDestructionMethod"

- Reflection: Added doc tag parsing support for properties (created T3P_Reflection_Property)
- Core: Implemented support for AOP Introduction declarations. Works, but needs a bit of a cleanup in T3P_Core_AOPFramework. Refs #35 and fixes #20.

#2 - 2007-03-01 19:38 - Robert Lemke

(In r160) * Core: Refactored the T3P_Core_AOPFramework: The builder functions are now in their own class and have been consolidated. Still needs some cleanup in the method interceptor code generator. Addresses #35.

#3 - 2007-03-04 18:06 - Robert Lemke

- *Status changed from New to Closed*
- *Sprint set to fixed*

Okay, I close this ticket for now - it's enough refactoring for the moment. If we feel like putting the code snippets into a template file, we can do so later on.

#4 - 2007-03-05 20:24 - Robert Lemke

(In r167) Refactored the T3P_Core_AOPProxyClassBuilder:

- Core: The AOPProxyClassBuilder smelled like Duplicated Code (76) and therefore I decided to split it up using the Template Method (325) pattern. Addresses #35.
- Core: Added tests and fixtures to assert that introducing multiple interfaces does work (it didn't work as my tests showed soon). Now everything seems to be fine and that fixes #37.

#5 - 2007-03-06 00:07 - Karsten Dambekalns

(In r169) Fixes failures in T3P_phpCRJackrabbit_CheckPermissionTest caused by incorrectly set up account permissions:

- repository.xml now does no longer provide a default user for logins without credentials
 - The unit test base class uses the configured "anonymous" user for read-only session access
- This fixes #35

Additionally T3P_phpCRJackrabbit_CheckPermissionTest contained some uses of + instead of . for string concatenation, this created invalid input.

#6 - 2007-03-06 00:11 - Karsten Dambekalns

Replying to [comment:5 karsten]:

| *(In r169) Fixes failures in T3P_phpCRJackrabbit_CheckPermissionTest caused by incorrectly set up account permissions:*

Sorry, wrong ticket id used in commit message. Can this be corrected?