TYPO3.Flow - Task # 2117

Status:	Resolved	**Priority:**	Should have
Author:	Robert Lemke	**Category:**	Core
Created:	2008-11-11	**Assigned To:**	Robert Lemke
Updated:	2011-08-04	**Due date:**	
Sprint:			
PHP Version:			
Has patch:			
Complexity:			
Subject:	Clean up bootstrap		

Description

The subpackages of FLOW3 are initialized in the bootstrap. Currently this whole process is hardcoded in the FLOW3 class which makes it impossible to deactivate certain subpackages if (not) needed. The range and order of subpackages should be configurable in the FLOW3.php configuration file

Related issues:

related to TYPO3.Flow - Feature # 3306: Flush routes cache automatically on c...	**Accepted**	**2009-05-14**

Associated revisions

Revision 132adbf5 - 2008-12-09 10:10 - Robert Lemke

Note: Although this commit contains the new File/Class Alteration Monitoring which is supposed to speed up FLOW3, a lot of speed optimizations are still missing and will follow during the next days.

- FLOW3: Started cleaning up the bootstrap. Addresses #2117
- FLOW3: Implemented a first version of a Class Alteration Monitoring which automatically flushes cache entries based on class reflection. Addresses #2284
- FLOW3: (Reflection) Refactored the Reflection Service so that information for one class can be flushed / rebuild individually instead of having to rebuild the whole reflection information. Resolves #2284
- FLOW3: (Cache) Fixed the *byTag() methods in the File Backend: If cache entry identifiers contained an underscore "_", the findByTag etc. didn't work any more. Now using "-" as an internal separator.
- FLOW3: (Utility) The temporary directory base is now set via a setter method instead of passing the whole FLOW3 settings to the constructor
- Testing: Adjusted the test runner to reflect the new bootstrap initialization methods

Revision a2e34b96 - 2008-12-09 10:10 - Robert Lemke

Note: Although this commit contains the new File/Class Alteration Monitoring which is supposed to speed up FLOW3, a lot of speed optimizations are still missing and will follow during the next days.

- FLOW3: Started cleaning up the bootstrap. Addresses #2117
- FLOW3: Implemented a first version of a Class Alteration Monitoring which automatically flushes cache entries based on class reflection. Addresses #2284
- FLOW3: (Reflection) Refactored the Reflection Service so that information for one class can be flushed / rebuild individually instead of having to rebuild the whole reflection information. Resolves #2284
- FLOW3: (Cache) Fixed the *byTag() methods in the File Backend: If cache entry identifiers contained an underscore "_", the findByTag etc. didn't work any more. Now using "-" as an internal separator.
- FLOW3: (Utility) The temporary directory base is now set via a setter method instead of passing the whole FLOW3 settings to the constructor
- Testing: Adjusted the test runner to reflect the new bootstrap initialization methods

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 catched 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

Revision 2cda63f5 - 2009-08-31 15:56 - Robert Lemke

[+FEATURE] FLOW3 (AOP): Parse errors in AOP proxy code (introduced by a potential bug) will now be treated properly by throwing a meaningful exception.

[-FEATURE] FLOW3 (Configuration): Removed support for PHP-based configuration files. This caused a lot of overhead and in practice we want everyone to use YAML files anyway.

[+FEATURE] FLOW3 (Configuration): Implemented support for cached ("compiled") configuration. When the new FLOW3.yaml option "configuration: comileConfigurationFiles" is turned on, all configuration will be compiled into a PHP file which from then on is read instead of parsing the YAML files everytime. Please see the updated documentation for more information. Resolves #4460

[~TASK] FLOW3 (Configuration): Completely refactored the Configuration Manager and a few related classes. Its methods now each fullfil a clear purpose and work more effective. The method names now also better describe what each method is doing. Resolves #2711

[-TASK] FLOW3 (Configuration): Removed the WritableSourceInterface - this is now a standard feature of a configuration source.

[+TASK] FLOW3 (Core): Refactored and cleaned up the FLOW3 Bootstrap. Constants are now defined by a static method and all configuration related functionality has been moved to the Configuration sub package. Also replaced the evil die() calls by more handsome exit() calls. Relates to #2117

[+TASK] FLOW3 (Object): Moved some object initialization code from the Bootstrap to the Object Manager and further cleaned up the Object Manager's code. Wrote additional tests - now this class has 100% test coverage. Relates to #2117

[+API] FLOW3 (MVC): Added a few @api annotations to properties of the AbstractController and ActionController.

[-API] FLOW3 (Utility): Removed the getSAPIType() method from the Environment class. Just use the constant FLOW3_SAPITYPE instead. Resolves #3567 and relates to #2117.

Revision 89c28b46 - 2009-08-31 15:56 - Robert Lemke

[+FEATURE] FLOW3 (AOP): Parse errors in AOP proxy code (introduced by a potential bug) will now be treated properly by throwing a meaningful exception.

[-FEATURE] FLOW3 (Configuration): Removed support for PHP-based configuration files. This caused a lot of overhead and in practice we want everyone to use YAML files anyway.

[+FEATURE] FLOW3 (Configuration): Implemented support for cached ("compiled") configuration. When the new FLOW3.yaml option "configuration: comileConfigurationFiles" is turned on, all configuration will be compiled into a PHP file which from then on is read instead of parsing the YAML files everytime. Please see the updated documentation for more information. Resolves #4460

[~TASK] FLOW3 (Configuration): Completely refactored the Configuration Manager and a few related classes. Its methods now each fullfil a clear purpose and work more effective. The method names now also better describe what each method is doing. Resolves #2711

[-TASK] FLOW3 (Configuration): Removed the WritableSourceInterface - this is now a standard feature of a configuration source.

[+TASK] FLOW3 (Core): Refactored and cleaned up the FLOW3 Bootstrap. Constants are now defined by a static method and all configuration related functionality has been moved to the Configuration sub package. Also replaced the evil die() calls by more handsome exit() calls. Relates to #2117

[+TASK] FLOW3 (Object): Moved some object initialization code from the Bootstrap to the Object Manager and further cleaned up the Object Manager's code. Wrote additional tests - now this class has 100% test coverage. Relates to #2117

[+API] FLOW3 (MVC): Added a few @api annotations to properties of the AbstractController and ActionController.

[-API] FLOW3 (Utility): Removed the getSAPIType() method from the Environment class. Just use the constant FLOW3_SAPITYPE instead. Resolves #3567 and relates to #2117.

History

#1 - 2009-05-19 12:21 - Robert Lemke

- Target version changed from 1.0 alpha 1 to 283

#2 - 2009-07-13 11:25 - Robert Lemke

- Target version deleted (283)

#3 - 2010-08-18 15:07 - Karsten Dambekalns

- Category set to Core

#4 - 2011-08-04 08:57 - Sebastian Kurfuerst

- Status changed from Accepted to Resolved

- % Done changed from 0 to 100

done