TYPO3.Flow - Feature # 5882

Status:	Closed	**Priority:**	Could have
Author:	Manuel Strausz	**Category:**	Package
Created:	2010-01-04	**Assigned To:**	
Updated:	2012-01-26	**Due date:**	
PHP Version:			
Has patch:	No		
Complexity:			
Subject:	Caching for Package Manager/Package file listing		
Description			

Currently, everytime FLOW3's Package Manager initializes it will traverse the folder structure via a DirectoryIterator. And for each active package, an array of class files will be built in much the same way.
This doesn't have much of a performance impact when deploying in a linux environment, however it seems to take about 0.5 - 1.0 seconds to do on a windows machine (at least that's the number I got from my tests).
I think the package manager and the packages themselves could really use some caching to solve this performance impact, and could generally be useful to minimize the number of filesystem operations. The difficult part about that is that at this point in the bootstrap process, there isn't any way to access the CacheManager (except by pre-loading it and circumventing the ObjectManager). So a different way of caching would be needed, maybe similarly to the way configuration files are cached.

History

#1 - 2010-01-13 16:40 - Robert Lemke

- Tracker changed from Bug to Feature

#2 - 2010-07-09 14:00 - Robert Lemke

- Target version set to 1.0 alpha 11

#3 - 2010-07-09 14:01 - Robert Lemke

- Category set to Package

- Priority changed from Should have to Could have

#4 - 2010-07-10 10:08 - Manuel Strausz

- File 5882.diff added

- % Done changed from 0 to 50

Since Robert recently changed this to target version alpha 11, I took the liberty of trying to implement this feature. :)

Provided is a patch which implements the described caching characteristics + caching the classFiles of all packages as well (which is where most of the performance can be gained by this optimization).
It basically works like this:

- write all package paths and the current package states into a serialized cache, when the package manager shuts down
- load the cache on the next initialization of the package manager, and check if the cache is still valid
- if any criteria is met that invalidates the cache, normal scanning of the packages will begin and the cache is rebuilt

The problem is invalidating the cache - since I can't analyze the directory structure, I am checking if the package states changed (at all) since the last cache was written. If it's in any way different, the cache will be invalid. Note that this works correctly for cases like creating a new package, but it won't register new/changed/deleted classes in the package. For this to work the cache file has to be deleted manually.
This is why this feature shouldn't be used in Development context, and is mostly suited for optimizing production environments. To activate package

path caching, the setting FLOW3.package.usePackagePathsCache needs to be set to "y" (default is is of course "n", for aforementioned reasons).

In a windows environment the performance improvement with a moderate amount of packages (FLOW3 framework packages + 2 custom ones) is about 200ms.

I tested this to the best of my abilities and it seems to work, but unfortunately I didn't quite see how to write a unit test for this. Since this is my first attempt at trying to provide a patch for FLOW3, I hope I got everything right according to the coding guidelines and that I didn't overlook anything. If there is anything wrong with the patch, please be so kind as to point it out to me so I can try to do better in the future.

I had to change the Package interface (a new method, setClassFiles) in order to enable class-filename caching, I hope you don't feel like I'm messing around too much in the guts of the framework with my first patch.

Also, if anyone has a better idea as to how to invalidate the cache (especially regarding the class file structure), I'd be happy to discuss and implement this. :)

#5 - 2010-07-10 10:34 - Manuel Strausz
- File 5882_fixed.diff added

Update: Had a small bug in the diff, which would lead to always writing the cache file even if it wasn't changed. Please use this diff instead of the one I posted above.

#6 - 2010-08-18 17:02 - Karsten Dambekalns
- Target version deleted (1.0 alpha 11)

#7 - 2012-01-26 15:24 - Robert Lemke
- Status changed from New to Closed
- Has patch set to No

Thanks a lot for the effort, Manuel. I took your suggestion and integrated it into the optimized class loading and bootstrapping mechanism for FLOW3 1.1

Files

5882.diff	10.9 kB	2010-07-10	Manuel Strausz
5882_fixed.diff	10.9 kB	2010-07-10	Manuel Strausz