TYPO3.Flow - Bug # 25965

Status:	Resolved	**Priority:**	Must have
Author:	Sebastian Kurfuerst	**Category:**	AOP
Created:	2011-04-14	**Assigned To:**	Karsten Dambekalns
Updated:	2011-04-19	**Due date:**	

PHP Version:	
Has patch:	
Complexity:	
Affected Flow version:	
Subject:	AOP ProxyClassBuilder always detects an aspect as "changed" if no Dependency Injection is used in this aspect (f.e. in LazyLoadingObjectAspect)

Description

Problem

If there is an aspect A, which does not need dependency injection, and **another** class C is modified, the ProxyClassBuilder is incorrectly assuming that Aspect A has changed, and triggers a full rebuild of **all** AOP target classes.

From method build() in Aop/Builder/ProxyClassBuilder:
1$rebuildEverything = FALSE;2foreach (array_keys($this->aspectContainers) as$aspectClassName) {3if ($this ->compiler->hasCacheEntryForClass($aspectClassName) === FALSE) {4$rebuildEverything = TRUE;5$this->systemLogger->log(sprintf('Aspect %s has been modified, therefore rebuilding all target classes.', $aspectClassName), LOG_INFO);6break;7   }8 }

for aspects which do not need Dependency Injection, "hasCacheEntryForClass" is always FALSE. This becomes clear when looking into the Object\Proxy\Compiler:
There, the cache is only written for classes where a proxy is needed.

Hotfix

It can be solved by adding empty Dependency Injection (though this is no real solution)... I am not sure how to solve this in a clean way, that's why I first wanted to document it here.

Related issues:		
related to TYPO3.Flow - Bug # 26107: Proxy class builder rebuild classes with...	**Resolved**	**2011-04-19**

Associated revisions

Revision b021a963 - 2011-04-15 12:32 - Karsten Dambekalns

[BUGFIX] Fix aspect change detection in AOP ProxyClassBuilder

If there is an aspect A, which does not need dependency injection, and

another class C is modified, the ProxyClassBuilder was incorrectly

assuming that aspect A had changed, and triggered a full rebuild of all

AOP target classes.

Change-Id: I4182a91bdf02e6ffd93804d2df76095f12bb4508
Fixes: #25965

History

#1 - 2011-04-14 16:44 - Sebastian Kurfuerst

!!! When fixing this issue, make sure to revert http://review.typo3.org/1614 again!

#2 - 2011-04-14 17:58 - Karsten Dambekalns

- Status changed from New to Accepted

- Assigned To set to Karsten Dambekalns

#3 - 2011-04-15 12:32 - Mr. Hudson

Patch set 1 of change I4182a91bdf02e6ffd93804d2df76095f12bb4508 has been pushed to the review server.

It is available at http://review.typo3.org/1633

#4 - 2011-04-15 12:33 - Karsten Dambekalns

- Status changed from Accepted to Under Review

#5 - 2011-04-19 11:30 - Karsten Dambekalns

- Status changed from Under Review to Resolved

- % Done changed from 0 to 100

Applied in changeset commit:b021a963cdcfc3b4e8cfbb07627a1a8cc64bf8ae.

#6 - 2011-04-19 11:32 - Christopher Hlubek

The change handling is still not optimal, see #26107.