Core - Bug # 53417

Status:	Rejected	**Priority:**	Should have
Author:	Viktor Livakivskyi	**Category:**	
Created:	2013-11-07	**Assigned To:**	
Updated:	2015-03-24	**Due date:**	
TYPO3 Version:	6.0		
PHP Version:	5.4		
Complexity:			
Is Regression:	No		
Sprint Focus:			
Subject:	ConfigurationManager->getConfigurationValueByPath() ignores AdditionalConfiguration.php		

Description

Hi.

Recently I've discovered an issue in ConfigurationManager. If I override something in AdditionalConfiguration.php, e.g. $GLOBALS['TYPO3_CONF_VARS']['BE']['compressionLevel'] = 4;, while in my LocalConfiguration.php I have this value set to 5, I'll get **4** in Install Tool, but following code will return **5**:

```
    $cm = \TYPO3\CMS\Core\Utility\GeneralUtility::makeInstance('TYPO3\\CMS\\Core\\Configuration\\ConfigurationManager');
    return $cm->getLocalConfigurationValueByPath('BE/compressionLevel');
```

Same result will be for ConfigurationManager->getConfigurationValueByPath(), which is used by Extension Manager and therefore I'm not able to override $GLOBALS['TYPO3_CONF_VARS']['EXT']['extConf'] for any extension, which makes it impossible to have different configurations for dev-stage-live environments.

Related issues:

related to Core - Bug # 57289: Respect additional configuration file for sile...	**Closed**	**2014-03-25**
duplicated by Core - Bug # 60650: ConfigurationManager::getConfigurationValue...	**Resolved**	**2014-07-29**

History

#1 - 2014-07-01 13:44 - Viktor Livakivskyi

Small update.
getLocalConfigurationValueByPath() may ignore AdditionalConfiguration.php, becasue it is explicitly mentioned in method's name, that it takes values form LocalConfiguration.php
but
getConfigurationValueByPath() should include defaultConfiguration merged with localConfiguration and merged with additionalConfiguration

#2 - 2014-07-08 17:06 - Markus Klein

@Christian: Can you maybe say something about this?

#3 - 2015-03-23 15:40 - Gerrit Code Review

- Status changed from New to Under Review

Patch set 1 for branch **master** of project **Packages/TYPO3.CMS** has been pushed to the review server.
It is available at http://review.typo3.org/38041

#4 - 2015-03-23 16:03 - Gerrit Code Review

Patch set 2 for branch **master** of project **Packages/TYPO3.CMS** has been pushed to the review server.
It is available at http://review.typo3.org/38041

#5 - 2015-03-23 16:32 - Mathias Brodala

IIRC this is on purpose since AdditionalConfiguration.php can contain arbitrary PHP code. This can break the install tool which is supposed to be the last resort in case of breakage and basically unbreakable.

#6 - 2015-03-24 14:47 - Viktor Livakivskyi

Mathias, you're right. But then ConfigurationManager->getConfigurationValueByPath() makes no sense to use, because AdditionalConfiguration.php is ignored. AdditionalConfiguration is intended to be used to override configuration from LocalConfiguration, and we need to rely on this in client code.

E.g. we have following in AdditionalConfiguration:

```
switch (\TYPO3\CMS\Core\Utility\GeneralUtility::getApplicationContext()) {
  case 'Development':
    $GLOBALS['TYPO3_CONF_VARS']['SYS']['displayErrors'] = 1;

    /* skip */

    break;
  case 'Production/Staging':
    $GLOBALS['TYPO3_CONF_VARS']['SYS']['displayErrors'] = -1;

    /* skip */

    break;
  case 'Production':
  default:
    $GLOBALS['TYPO3_CONF_VARS']['SYS']['displayErrors'] = 0;

    /* skip */

}
```

And if I get ConfigurationManager->getConfigurationValueByPath('SYS/displayErrors') I'll not get the value, dependant on context, but one, defined in LocalConfiguration.php
So, I can't rely on this value.

#7 - 2015-03-24 14:51 - Markus Klein

ConfigurationManager is internal API, see class header

> - *IMPORTANT:*
> - *This class is intended for internal core use ONLY.*

#8 - 2015-03-24 15:01 - Viktor Livakivskyi

Why is it then mentioned in "What's New" slides of TYPO3 6.0 on page 35 in "Coding Examples"?

http://forge.typo3.org/attachments/download/24568/TYPO3-v6-0-whats-new.pdf

Maybe, makes sense to add remark in that PDF, that developes shouldn't use it or completely remove that page?

UPDATE: also checked code for 6.0 and 6.1 - there were no remark about @internal, therefore I suggested it's usage in client code at the moment of creating this issue

#9 - 2015-03-24 15:08 - Markus Klein

Ok that is really bad. Although the title of the slide talks about "LocalConfiguration".
I doubt anybody will touch the slides, since 6.0 is not supported for a long time now.

#10 - 2015-03-24 15:15 - Viktor Livakivskyi

Markus, you're right.
Since 6.2 marks this class as internal, then it makes no sense to use it in client code.

So, the issue may be closed.

#11 - 2015-03-24 15:23 - Christian Kuhn

- Status changed from Under Review to Rejected

Hey.

It was unfortunate that the internal low level class to handle LocalConfiguration was mentioned in the slides of former core versions. We clarified with the @internal annotation only later and now in result stumble upon the one or other one complaining about the situation. Sorry about that.

The issue however is bogus since AdditionalConfiguration is not handled at this point by intention - otherwise settings from additionalconfiguration could end up in localconfiguration on writing, which is not wanted.

All "using / 3rd party code" should operate (reading) on $GLOBALS['TYPO3_CONF_VARS'] directly and will then have the "final" array.

This issue will be closed as rejected now.