TYPO3.Flow - Feature # 10524

Status:	Closed	**Priority:**	-- undefined --
Author:	Martin Eisengardt	**Category:**	
Created:	2010-10-29	**Assigned To:**	
Updated:	2012-05-15	**Due date:**	

PHP Version:	
Has patch:	No
Complexity:	
Subject:	Support multiple package directories

Description

Currently I have trouble developing several seperated modules. To test them each module must either be starting at the local Packages/<something> folder or must contain a copy/reference to the whole framework. That's blowing up my eclipse workspace.

Flow3 should support (at least during development) the directory structure:

```
/workspace-root/framework-project/Packages/Framework/DocumentationBrowser ...
/workspace-root/framework-project/Packages/Framework/ExtJS ...
/workspace-root/framework-project/Packages/Framework/FLOW3 ...
/workspace-root/framework-project/Packages/Framework/FLUID ...
[etc....]
/workspace-root/my-project/Configuration...
/workspace-root/my-project/Data...
/workspace-root/my-project/Packages/MyLib/MyModuleName...
/workspace-root/my-project/Web/index.php
/workspace-root/another-project/Configuration...
/workspace-root/another-project/Data...
/workspace-root/another-project/Packages/MyLib/MyModuleName...
/workspace-root/another-project/Web/index.php
```

- Benefits from this structure
  - Eclipse does only have to manage the framework components once
  - There is no need to prepare a more complex layout over svn/git
  - There may be other use cases where multiple projects are joined together to create an application

- Changes that need to be made
  The changes are simple.
  1) The defines FLOW3_PATH_CONFIGURATION and FLOW3_PATH_DATA need to be overwritten by htaccess environment variables. Currently Bootstrap does declare them depending on FLOW3_ROOTPATH. In the directory layout above the configuration and data directories are part of the project and not part of the flow3 root.
  2) (already possible) FLOW3_ROOTPATH will be set in the htaccess.
  3) A new (but optional) environment variable/define FLOW3_PATH_PACKAGES_ADDITIONAL will contain a list of additional directories where to find packages. htacces or the index.php script will have to declare the directories where to find the additional package directories. This will cause the following changes in PackageManager:

```
/**
 * Scans all directories in the packages directories for available packages.
 * For each package a \F3\FLOW3\Package\ object is created and returned as
 * an array.
```

```
 *
 * @return void
 * @author Robert Lemke <robert@typo3.org>
 */
protected function scanAvailablePackages() {
    $this->packages = array('FLOW3' => $this->objectManager->create('F3\FLOW3\Package\Package', 'FLOW3',
FLOW3_PATH_FLOW3));

    $this->scanPathForPackages(FLOW3_PATH_PACKAGES);
    if (defined('FLOW3_PATH_PACKAGES_ADDITIONAL')) {
        foreach (explode(';', FLOW3_PATH_PACKAGES_ADDITIONAL) as $path) {
            $this->scanPathForPackages($path);
        }
    }

    foreach (array_keys($this->packages) as $upperCamelCasedPackageKey) {
        $this->packageKeys[strtolower($upperCamelCasedPackageKey)] = $upperCamelCasedPackageKey;
    }
}

/**
 * Scans a directory for packages.
 *
 * some of the code was taken from flow3
 *
 * @param string $path
 *
 * @throws \F3\FLOW3\Package\Exception\DuplicatePackageException
 */
protected function scanPathForPackages($path)
{
    foreach (new \DirectoryIterator($path) as $parentFileInfo) {
        $parentFilename = $parentFileInfo->getFilename();
        if ($parentFilename[0] === '.' || !$parentFileInfo->isDir()) continue;

        foreach (new \DirectoryIterator($parentFileInfo->getPathname()) as $childFileInfo) {
            $childFilename = $childFileInfo->getFilename();
            if ($childFilename[0] !== '.' && $childFilename !== 'FLOW3') {
                $packagePath = \F3\FLOW3\Utility\Files::getUnixStylePath(realpath($childFileInfo->getPathName())) . '/';
                if (isset($this->packages[$childFilename])) {
                    throw new \F3\FLOW3\Package\Exception\DuplicatePackageException('Detected a duplicate package, remove
either "' . $this->packages[$childFilename]->getPackagePath() . '" or "' . $packagePath . '".', 1253716811);
                }
                $this->packages[$childFilename] = $this->objectManager->create('F3\FLOW3\Package\Package',
$childFilename, $packagePath);
            }
        }
    }
}
```

History

#1 - 2012-03-12 18:35 - Christian Müller

- Status changed from New to Needs Feedback

- Has patch set to No

What speaks against symlinking the Framework directory to your projects?

#2 - 2012-05-15 15:39 - Robert Lemke

- Status changed from Needs Feedback to Closed

- Priority changed from Should have to -- undefined --

Nothing more happened with this issue and it's unlikely that we need it considering the possibility to use symlinks and the planned integration of Composer.