TYPO3.Fluid - Bug # 38369

Status:	New	**Priority:**	Must have
Author:	Bastian Waidelich	**Category:**	View
Created:	2012-06-25	**Assigned To:**	
Updated:	2012-06-25	**Due date:**	

Has patch:	No
Affected Flow version:	Git 1.1
Subject:	Resource ViewHelpers should not fall back to request package

Description

Currently the f:uri.resource ViewHelper uses the package key of the current request to determine the absolute resource path, if no package is specified.

The resource interceptor that replaces links to assets by the resource viewhelper, so:

```
1<link rel="stylesheet" href="../../../Public/Stylesheets/SomeFile.css" media="all" />
```

is (internally) converted into

```
1<link rel="stylesheet" href="{f:uri.resource(resource: 'Stylesheets/SomeFile.css')}" media="all" />
```

So the resource ViewHelper uses the **current** package key to resolve the path to:
Current/Package/Resources/Public/Stylesheets/SomeFile.css.
While this mostly works for FLOW3, in Phoenix the current package is usually **TYPO3.TYPO3** and resources are located in some site package.

Therefore the TypoScript Fluid renderer sets the requests package key from the Fluid template path:

```
1if (strpos($templatePath, 'resource://') === 0) {
2    $tmp = substr($templatePath, 11);
3    $tmp2 = explode('/', $tmp);
4
5    $fluidTemplate->getRequest()->setControllerPackageKey(array_shift(($tmp2)));
6}
```

This has the side effect, that all links in the Fluid template now point to the site package instead of the FrontendNode controller of the TYPO3 package!

I didn't set the category to ViewHelpers because IMO this should be fixed in the view: We probably need something like resourcesPackage or similar that should be used by the resource ViewHelper instead of the requests package key.

History

#1 - 2012-06-25 15:50 - Bastian Waidelich

Bastian Waidelich wrote:

> I didn't set the category to ViewHelpers because IMO this
> should be fixed in the view

BTW: This would be a good chance to clean up the code in the FluidRenderer (:

```
1 if (strpos($templatePath, 'resource://') === 0) {
2     $resourcePathWithoutProtocol = substr($templatePath, 11);
3     $resourcePathSegments = explode('/', $resourcePathWithoutProtocol);
4     $resourcePackageKey = array_shift($resourcePathSegments);
5     $fluidTemplate->setDefaultResourcePackage($resourcePackageKey);
6 }
```

#2 - 2012-06-25 15:55 - Karsten Dambekalns

IMHO the description is misleading, as this is not an issue with the RVH, but rather a "usage problem", in that for Phoenix there is a difference between "current and current" ;).