

TYPO3.Flow - Feature # 33937

Status:	Accepted	Priority:	Should have
Author:	Bastian Waidelich	Category:	Resource
Created:	2012-02-15	Assigned To:	Karsten Dambekalns
Updated:	2014-04-15	Due date:	
PHP Version:			
Has patch:	No		
Complexity:			
Subject:	Convenience method to resolve public "resource://" paths		
Description			
<p>It would be helpful to have an easy way to resolve the 'resource://...' path to static resources.</p> <p>Currently we end up using code like the following a lot:</p> <pre>protected function resolveResourcePath(\$resourcePath) { \$matches = array(); preg_match('#resource://(?:[^/]*)/Public/(.*)#', \$resourcePath, \$matches); if (\$matches === array()) { throw new \TYPO3\Fluid\Core\ViewHelper\Exception("Resource path '" . \$resourcePath . "' can't be resolved.", 1328543327); } \$packageKey = \$matches[1]; \$path = \$matches[2]; return \$this->resourcePublisher->getStaticResourcesWebBaseUri() . 'Packages/' . \$packageKey . '/' . \$path; }</pre>			

History

#1 - 2012-02-16 11:19 - Karsten Dambekalns

- Status changed from New to Needs Feedback
- Assigned To set to Karsten Dambekalns

What is this needed for? Any file functions can use the resource:// URL directly. To fetch the web URL, a method exists already (no?).

#2 - 2012-03-07 16:15 - Bastian Waidelich

Ok, here's the ViewHelper which needed this:

```
1<?php
2namespace TYPO3\Form\ViewHelpers;
3
4/*
5 * This script belongs to the FLOW3 package "TYPO3.Form".
6 *
7 * It is free software; you can redistribute it and/or modify it under
8 * the terms of the GNU Lesser General Public License, either version 3
```

```

9 * of the License, or (at your option) any later version.      *
10 *                                                              *
11 * The TYPO3 project - inspiring people to share!              *
12 *                                                            */
13
14use TYPO3\FLOW3\Annotations as FLOW3;
15
16/**
17 * Output the configured stylesheets and JavaScript include tags for a given preset
18 */
19class RenderHeaderViewHelper extends \TYPO3\FLOW3\Core\ViewHelper\AbstractViewHelper {
20
21 /**
22 * @FLOW3\Inject
23 * @var \TYPO3\FLOW3\Resource\Publishing\ResourcePublisher
24 */
25 protected $resourcePublisher;
26
27 /**
28 * @FLOW3\Inject
29 * @var \TYPO3\FLOW3\Form\Factory\ArrayFormFactory
30 */
31 protected $formBuilderFactory;
32
33 /**
34 * @param string $presetName name of the preset to use
35 * @return string the rendered form head
36 */
37 public function render($presetName = 'default') {
38     $content = "";
39     $presetConfiguration = $this->formBuilderFactory->getPresetConfiguration($presetName);
40     $stylesheets = isset($presetConfiguration['stylesheets']) ? $presetConfiguration['stylesheets'] : array();
41     foreach ($stylesheets as $stylesheet) {
42         $content .= sprintf('<link href="%s" rel="stylesheet">', $this->resolveResourcePath($stylesheet['source']));
43     }
44     $javaScripts = isset($presetConfiguration['javaScripts']) ? $presetConfiguration['javaScripts'] : array();
45     foreach ($javaScripts as $javaScript) {
46         $content .= sprintf('<script src="%s"></script>', $this->resolveResourcePath($javaScript['source']));
47     }
48     return $content;
49 }
50
51 /**
52 * @param string $resourcePath
53 * @return string
54 */
55 protected function resolveResourcePath($resourcePath) {
56     // TODO: This method should be somewhere in the resource manager probably?
57     $matches = array();
58     preg_match('#resource://(?:[/]*)/Public/(.*)#', $resourcePath, $matches);
59     if ($matches === array()) {
60         throw new \TYPO3\FLOW3\Core\ViewHelper\Exception('Resource path "' . $resourcePath . '" can\'t be resolved.', 1328543327);
61     }

```

```
62  $package = $matches[1];
63  $path = $matches[2];
64  return $this->resourcePublisher->getStaticResourcesWebBaseUri() . 'Packages/' . $package . '/' . $path;
65  }
66}
67?>
```

`$stylesheet['source']` and `$javascript['source']` contains paths like "resource://SomePackage/foo/bar".

If I see it correctly a similar code is used in the `uri.resource ViewHelper` but there should be some convenience method in the `resource manager/publisher` probably don't you think?

#3 - 2012-03-07 16:22 - Karsten Dambekalns

- *Status changed from Needs Feedback to Accepted*

#4 - 2012-08-14 10:33 - Karsten Dambekalns

- *Target version set to 2.0 beta 1*

#5 - 2012-10-16 16:38 - Marc Neuhaus

Another use-case:

i'm writing a Surf Deployment to push a file from inside a package to the remote node and need a proper way to have a real path to a resource file to use it with a shell command :)

#6 - 2012-12-10 13:37 - Karsten Dambekalns

- *Target version changed from 2.0 beta 1 to 2.1*

#7 - 2013-05-21 13:28 - Robert Lemke

- *Target version deleted (2.1)*

#8 - 2014-04-15 11:40 - Christian Müller

I guess this won't be implemented as API method. We can reconsider after the new resource mgm is merged. It could be a non API public method in the local file system implementation.