

TYPO3.Fluid - Feature # 54155

Status:	Rejected	Priority:	Could have
Author:	Kay Strobach	Category:	ViewHelpers
Created:	2013-12-03	Assigned To:	Bastian Waidelich
Updated:	2013-12-06	Due date:	
Has patch:	No		
Subject:	f:render partial="" package=""		
Description	<p>Hi guys,</p> <p>especially when developing Applications, which rely on an AbstractController, which sets the layout it would be nice to have a way to use the centralized defined partials and the one from each package.</p> <p>Therefore it would be nice to have a package attribute for finding the related partial.</p>		
Related issues:			
related to TYPO3.Fluid - Feature # 50826: add resource as parameter for <f:re...		Closed	2013-08-06
duplicates TYPO3.Fluid - Feature # 33487: Configure from what package to use ...		Closed	2012-01-27

History

#1 - 2013-12-03 13:14 - Bastian Waidelich

- Status changed from New to Rejected
- Assigned To set to Bastian Waidelich
- Priority changed from Must have to Could have

Hey Kay,

thanks for the feature request, but this is already possible with the [Views.yaml configuration](#)

#2 - 2013-12-03 13:16 - Bastian Waidelich

Example (untested):

- 1-
- 2 requestFilter: 'isPackage("Some.Package") && isController("Some")'
- 3 options:
- 4 partialRootPaths: ['resource://Other.Package/Private/Partials']

#3 - 2013-12-03 13:25 - Kay Strobach

following situation:

- layout uses partials from the root package.
- Controller uses partials from the package where it resides.

This is also interesting for the bootstrap package ... AFAIK this is NOT possible - or do I have to add many paths?

Then this can be changed into a Documentation Issue ;)

#4 - 2013-12-06 13:20 - Rafael Kähm

for using partials across packages add in your packages, which supply other Packages with partials, Views.yaml with following stuff:

```
-
  options:
    partialRootPaths:
      'You.Package/Partials': 'resource://You.Package/Private/Partials'
```

If your partial has unique name Fluid will find this partial. If you have multiple partials in different packages with same name, then put the partials in subfolders to facilitate finding a required partial:

```
<f:render partial="SubfolderFromPartialRootPath/NavBarLoginBox" />
```

Note: Fluid uses first partial, which it finds.

#5 - 2013-12-06 13:57 - Bastian Waidelich

Rafael Kähm wrote:

```
| If your partial has unique name Fluid will find this partial. If you have multiple partials in different packages with same name, then put the partials in
| subfolders to facilitate finding a required partial:
| 'You.Package/Partials': 'resource://You.Package/Private/Partials'
```

This example might be a bit misleading:

While it's a good idea to use named array indexes (so that it can be overridden more easily by 3rd party packages) the array key does not really matter.

@Kay With the "supported options" introduced with <https://review.typo3.org/16393> I think all common scenarios can be covered.

"partialRootPaths" is an array in order to allow you to create arbitrary fallbacks:

```
1-
2 requestFilter: 'isPackage("Some.Package") && isController("Some")'
3 options:
4 partialRootPaths:
5 'Some.Package': 'resource://Some.Package/Private/Partials'
6 'Other.Package': 'resource://Other.Package/Private/Partials'
```

Would always first look in Some.Package and fallback to Other.Package if a partial wasn't found. With the requestFilter you can decide for which requests this fallback rule should apply.

NOTE: To many of those complex rules can make your application very hard to grasp! I usually prefer to keep all templates, layouts & partials in one place - for 3rd party packages I copy them to my App package. This has the drawback that you need to sync changes when you update, but I prefer to do this explicitly anyways

