

TYPO3.Fluid - Feature # 50801

Status:	Closed	Priority:	Could have
Author:	Rafael Kähm	Category:	View
Created:	2013-08-05	Assigned To:	Bastian Waidelich
Updated:	2013-08-08	Due date:	
Has patch:	No		
Subject:	Arguments "path" and "package" for f:layout View Helper		
Description	<p>This is usefull if you use multiple packages for the same application.</p> <p>Make sure you use "package" argument for each VH that needs this argument inside used layout, because Fluid does not know "origin" (also where layout come from) and tries to fetch controller and action from package, which uses this layout.</p> <p>Also next step is to implement Node-"origin" functionality and after that Arguments "path" and "package" for f:render View Helper (for Partials).</p> <p>gerrit review comes tomorrow.</p>		

History

#1 - 2013-08-06 13:48 - Gerrit Code Review

- Status changed from New to Under Review

Patch set 1 for branch **master** has been pushed to the review server.

It is available at <https://review.typo3.org/22839>

#2 - 2013-08-06 14:01 - Gerrit Code Review

Patch set 2 for branch **master** has been pushed to the review server.

It is available at <https://review.typo3.org/22839>

#3 - 2013-08-06 14:17 - Bastian Waidelich

- Status changed from Under Review to Needs Feedback

Thanks for the feature request and especially for the patch. But I'm afraid we won't need such feature once <https://review.typo3.org/#/c/16392/> is merged

#4 - 2013-08-06 15:46 - Adrian Förder

Hi Bastian; how would you do that with the patch you've mentioned?

#5 - 2013-08-06 15:55 - Bastian Waidelich

Adrian Förder wrote:

| *Hi Bastian; how would you do that with the patch you've mentioned?*

Create a **Views.yaml** and override the default layoutRootPath:

```
1 -
2 options:
3   layoutRootPath: 'resource://Some.Package/Private/Layouts'
```

If you only want to override it for certain packages/controllers/actions you can use the **requestPattern** configuration

#6 - 2013-08-06 18:35 - Rafael Kähm

Hi Bastian,

i see no opposites to using both patches for that. #50801 and #50826 are backward compatible to current Fluid and i think it is similarly compatible to #42176.

#7 - 2013-08-07 11:52 - Adrian Förder

Hi Bastian again,

but still with Views.yaml you can only set the path, not the complete "url" to the file.

Consider a use case where you have a Neos Site Package and a plain Flow package and you want to cross-reference concrete partial files, for example, just to avoid duplicate template code.

#8 - 2013-08-07 12:02 - Bastian Waidelich

Rafael Kähm wrote:

| *i see no opposites to using both patches for that.*

The patches are compatible, yes. But we prefer a single way of configuration and I can't think of a scenario that is possible with this patch but not with the Views.yaml change

Adrian Förder wrote:

| *but still with Views.yaml you can only set the path, not the complete "url" to the file.*

Of course:

```
1 -
2 requestFilter: 'isPackage("Original.Package") && isController("Foo") && isAction("bar")'
3 options:
4   partialPathAndFilenamePattern: 'resource://Some.Package/Private/Partials/Foo.html'
```

Or did I get you wrong?

I agree, setting "path"/"package" on the layout view helper is simpler for that use case. But with the views configuration we have one way of covering all sorts of scenarios.

#9 - 2013-08-07 12:22 - Adrian Föder

- Status changed from Needs Feedback to On Hold
- Priority changed from Must have to Could have

Bastian, for the mentioned configuration you will force-overwrite the whole partial path and file name (you even don't make it a "pattern" but a fixed value), so, different partials along the same Action request won't be possible, no?

```
1Hello <f:render partial="Party/NameAndIcon" />,  
2here's your map: <f:render partial="Party/LocationMap" />
```

won't this always result into 'resource://Some.Package/Private/Partials/Foo.html' for both partials as configured in your example?

#10 - 2013-08-07 13:01 - Bastian Waidelich

Adrian Föder wrote:

| Bastian, for the mentioned configuration you will force-overwrite the whole partial path and file name

Yes, you're right. This probably makes more sense with layouts. However with the "Fallback path" feature (#39870) it should be possible to do something like:

```
1 -  
2 requestFilter: 'isPackage("Original.Package") && isController("Foo") && isAction("bar")'  
3 options:  
4 partialRootPaths: ['resource://Some.Package/Private/Partials/', 'resource://Original.Package/Private/Partials/']
```

I really don't want to be too restrictive and maybe I'm missing something, but I really fear Fluid getting too complex (code and usage-wise). Let's see what others think about this

#11 - 2013-08-08 13:20 - Bastian Waidelich

- Status changed from On Hold to Closed
- Assigned To set to Bastian Waidelich

Closing in favor of #42176 (as discussed via IRC)