

TYPO3.Flow - Feature #7113

Make view resolution in ActionController independent from Fluid

2010-04-02 11:11 - Christopher Hlubek

Status:	Resolved	Start date:	2010-04-02
Priority:	Should have	Due date:	
Assignee:	Robert Lemke	% Done:	100%
Category:	MVC	Estimated time:	0.00 hour
Target version:	1.0 alpha 11	Complexity:	
PHP Version:			
Has patch:			

Description

Currently the view for any ActionController is tied to Fluid. For every action processed, a Fluid TemplateView will be created. Only if a template cannot be found the format is taken into account for the view object name.

This will work out as long as there is no Fluid template for an action that should be rendered with another view. Considering the case that we need format specific views (JSON, PDF, CSV, ...) for the *same* action that also renders HTML we need to change the current mechanism.

I would propose a priority based resolver like used for the request handlers. So a view would then be resolved by a view resolver that checks if a view implementation can handle the current request (most commonly by checking the format).

Associated revisions

Revision 5022 - 2010-08-19 17:50 - Robert Lemke

[~TASK] TYPO3 (Controller): Adjusted controller to new Action Controller with multi-format-support. Relates to #7113

Revision 5023 - 2010-08-19 17:55 - Robert Lemke

[~TASK] FLOW3 (MVC): Changed the order how views are resolved: If a dedicated view exists for the controller / action pair, it is chosen instead of the Fluid Template View.

[+FEATURE] FLOW3 (MVC): The ActionController now allows for specifying view object names for each format, e.g. a specific view for JSON request, another one for HTML etc. Resolves #7113

[!!!][API] FLOW3 (MVC): Added a new method to the ViewInterface: canRender() must tell if the view is able to render something meaningful for the given controller context.

[~TASK] Fluid (View): Adjusted the template view to the extended ViewInterface.

Revision 4d203b18 - 2010-08-23 17:12 - Robert Lemke

[~TASK] FLOW3 (MVC): Changed the order how views are resolved: If a dedicated view exists for the controller / action pair, it is chosen instead of the Fluid Template View.

[+FEATURE] FLOW3 (MVC): The ActionController now allows for specifying view object names for each format, e.g. a specific view for JSON request, another one for HTML etc. Resolves #7113

[!!!][API] FLOW3 (MVC): Added a new method to the ViewInterface: canRender() must tell if the view is able to render something meaningful for the given controller context.

[~TASK] Fluid (View): Adjusted the template view to the extended ViewInterface.

Change-Id: I36517e1fbd70d26c049e33ac662a5d8ae686d278

History

#1 - 2010-04-16 14:53 - Sebastian Kurfuerst

I'll be available for such discussions

#2 - 2010-07-09 14:41 - Robert Lemke

- Assignee set to Christopher Hlubek

- Target version set to 1.0 alpha 11

Okay, I think a decision based on the format would be okay. How about defining that directly in the Controller in form of an array?

```
$formatToViewObjectNameMap = array(
    '*' => 'F3\Fuild\View\TemplateView',
    'json' => '....MyOwnView',
```

```
..  
);
```

could support regex.

Even if you don't like my suggestion, just start working on it ... you'll know how.

#3 - 2010-08-11 11:59 - Robert Lemke

- Status changed from New to Accepted

- Assignee changed from Christopher Hlubek to Robert Lemke

#4 - 2010-08-19 18:00 - Robert Lemke

- Status changed from Accepted to Resolved

- % Done changed from 0 to 100

Applied in changeset r5023.

#5 - 2010-08-20 11:01 - Julian Kleinhans

I think there is something wrong..

since yesterday, after update the latest trunk i get a 404 with this error message from the action controller. "Sorry, the page you requested was not found. The technical reason is: No template was found. View could not be resolved for action "index".

Is there any change in the conventions ?

I have a StandardController with a indexAction method and i have a template in Resources/Private/Templates/Standard/index.html

any ideas ?

#6 - 2010-10-14 12:23 - Franz Kugelmann

Hi Julian,

did you try with the Template-name in UpperCase (Index.html)?

At least on non-Windows that makes a big difference :)

If i remember correctly, that was necessary at some point some months ago.

Julian Kleinhans wrote:

I think there is something wrong..

since yesterday, after update the latest trunk i get a 404 with this error message from the action controller. "Sorry, the page you requested was not found. The technical reason is: No template was found. View could not be resolved for action "index".

Is there any change in the conventions ?

I have a StandardController with a indexAction method and i have a template in Resources/Private/Templates/Standard/index.html

any ideas ?