

TYPO3.Fluid - Feature # 27119

Status:	Resolved	Priority:	Should have
Author:	Fernando Arconada	Category:	Core
Created:	2011-05-29	Assigned To:	Bastian Waidelich
Updated:	2013-01-10	Due date:	
Has patch:	No		
Subject:	Declare namespaces in XML tag		
Description	One of the great things of FLUID is that is XML but namespace declaration isnt. There should be a way/tag to declare namespaces in a tag. If you could have your template as pure XML you could validate it with XML validation tools		
Related issues:	related to Extension Builder - Task # 65547: Generate Fluid templates with fl...		Needs Feedback 2015-03-05

Associated revisions

Revision 432ebf65 - 2012-05-08 16:20 - Bastian Waidelich

[FEATURE] Declare namespaces in XML tag

This makes it possible to declare Fluid namespace identifiers with the common xmlns syntax `<html xmlns:foo="http://some/unique/namespace">` That works as a replacement for xml/html based Fluid templates which can be used instead of the `{namespace foo=Bar}` syntax.

This syntax triggers auto-completion in most IDEs if the xml namespace is bound to a valid XSD schema!

Usage:

- Create a XSD file for your ViewHelpers by executing
`./flow3 documentation:generatexsd Your\Package\ViewHelpers --target-file some/directory/your.package.xsd`
- Import the XSD file in your favorite IDE and map it to the namespace `"http://typo3.org/ns/Your/Package/ViewHelpers"`
- Add the namespace to your Fluid template by adding the "xmlns:" attribute to the root tag (usually `<xml ...>` or `<html ...>`).

If you want to use this inside partials, you can use the "section" argument of the render ViewHelper in order to only render the content of the partial:

Partial:

```
<html xmlns:x="http://typo3.org/ns/Your/Package/ViewHelpers">
<f:section name="content">
  <x:yourViewHelper />
</f:section>
```

Template:

```
<f:render partial="PartialName" section="content" />
```

Note:

You are able to use a different XML namespace pattern by specifying the `--xsd-namespace` argument in the `generatexsd` command.

The default pattern `"http://typo3.org/ns/<php namespace>"` is resolved automatically by the Fluid parser. If you use a custom XML namespace, you need to configure the namespace mapping.

The (YAML) syntax for that is:

TYPO3:

Fluid:

namespaces:

```
'http://some/unique/namespace': 'My\Php\Namespace'
```

Change-Id: I4d184ee81677c59583bdc7f6f19375155b60efe8

Resolves: #27119

Releases: 1.1

History

#1 - 2012-02-10 11:25 - Adrian Förder

+1 on that; I ever wondered why this wasn't regarded initially. Are/were there technical / parsing reasons for that?

#2 - 2012-03-05 18:44 - Bastian Waidelich

- *Category set to Core*

Good point! We might go for the XML namespace syntax (`<html [...] xmlns:foo="bar">`).

But we would need to map the XML namespace to the PHP Namespace somehow and this has to be global!

#3 - 2012-03-05 19:23 - Claus Due

Suggestion:

If we are willing to make a breaking change, we can force that the XML namespace used must match the package/extension key that contains the corresponding ViewHelpers.

Such behavior could perhaps also be toggled on a per-package/extension basis through settings, making it non-breaking?

I would love to see this done!

#4 - 2012-03-05 19:52 - Bastian Waidelich

Claus Due wrote:

Hi Claus,

If we are willing to make a breaking change, we can force that the XML namespace used must match the package/extension key that contains the corresponding ViewHelpers.

Actually the XML namespace **identifier** does not have to be unique and I prefer to have a short identifier here instead of a full extension key. We "only" need to have a unique mapping from the actual namespace URI to the ViewHelper path, so if you have:

```
1<html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en" xmlns:foo="http://bar.tld/ns/my/ViewHelpers" ... />
```

Fluid needs to know where ViewHelpers with the namespace "http://bar.tld/ns/my/ViewHelpers" are located. This should be backwards compatible.

#5 - 2012-03-05 19:57 - Claus Due

Hi Bastian,

You're right, there's also a point in registering different VH sets under the same namespace in different templates.

How about this, which should also be 100% backwards compatible:

```
<html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en" xmlns:foo="http://bar.tld/ns/my/ViewHelpers">
  <f:namespace xmlns="foo" namespace="\Foo\Bar\ViewHelpers" />
</html>
```

In this case you could use either the {namespace} annotation or the <f:namespace> - and it would only be loosely connected to the <html xmlns:foo> definition. Your thoughts?

EDIT: since there are no valid tags we can apply to <html> and we cannot yet in that hierarchy level use <f:...> I believe the nested <f:namespace> ViewHelper is a good solution.

#6 - 2012-03-07 12:35 - Bastian Waidelich

Claus Due wrote:

Hi Claus,

In this case you could use either the {namespace} annotation or the <f:namespace> - and it would only be loosely connected to the <html xmlns:foo> definition. Your thoughts?

Yes, I thought about a tag based syntax for {namespace} too and I think this is feasible. But IMO it would be even better if you would not have to define the namespace twice. And if we make it possible to use either {namespace} or <html|xml xmlns:foo="bar" (with a preference to the tag based syntax for XHTML/XML based templates) users would get auto completion by default.. I'll give that a try!

#7 - 2012-03-07 17:37 - Bastian Waidelich

- Status changed from New to Accepted
- Assigned To set to Bastian Waidelich

#8 - 2012-03-07 17:41 - Bastian Waidelich

Ok I've done a quick prototype that works quite nicely for me.

You'll have to "register" namespace mappings in the global configuration. For FLOW3 in Configuration/Settings.yaml like:

```
1TYPO3:
2 Fluid:
3 namespaces:
4 'http://typo3.org/ns/typo3/fluid/viewhelpers': 'TYPO3\Fluid\ViewHelpers'
5 'http://typo3.org/ns/typo3/viewhelpertest/viewhelpers': 'TYPO3\Viewhelpertest\ViewHelpers'
```

And then you can import those namespaces in your Fluid Template like:

```
1<?xml version="1.0" encoding="utf-8" ?>
2<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
3<html xmlns="http://www.w3.org/1999/xhtml"
4 xmlns:f="http://typo3.org/ns/typo3/fluid/viewhelpers"
5 xmlns:vht="http://typo3.org/ns/typo3/viewhelpertest/viewhelpers">
6
7 <f:layout name="Default" />
8 <f:section name="content">
9 <vht:meHasAutoCompletion rocks="true" />
10 </f:section>
11</html>
```

Please comment

#9 - 2012-03-07 17:46 - Gerrit Code Review

- Status changed from Accepted to Under Review

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

It is available at <http://review.typo3.org/9463>

#10 - 2012-03-07 17:49 - Gerrit Code Review

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

It is available at <http://review.typo3.org/9463>

#11 - 2012-03-07 17:50 - Gerrit Code Review

Patch set 3 for branch **master** has been pushed to the review server.

It is available at <http://review.typo3.org/9463>

#12 - 2012-03-07 18:06 - Adrian Förder

Awesome; how could I overlook such a hot topic...

I have a single demur; it's about the namespace declaration at all:

```
<html xmlns:foo="http://some/unique/namespace">
```

this requires to be set in every file; and it's semantically incorrect: with this you introduce a new root node for each file, which is even `<html>` and could heavily be confused with the actual `<html>` element. Is it even put out?

Or is your intention to set the namespace via `<html>` and "ship" everything out also, including inheriting the setting to partials and layout-children? That would destroy the idea of local namespacing.

#13 - 2012-03-07 18:14 - Adrian Förder

aandd, I would place the configuration to something including "ViewHelpers", using only

```
TYPO3:  
  Fluid:  
    namespaces:
```

could be too-easily mixed up with some real PHP class based namespace stuff.. so I'd suggest at least some kind of

```
TYPO3:  
  Fluid:  
    viewHelpers:  
      namespaces:
```

or

```
viewHelperNamespaces:
```

#14 - 2012-03-07 19:51 - Bastian Waidelich

@Adrian good points. If you look at my example, the `html` tag is actually outside the rendered content. You can do the same in partials by using inline sections. Re configuration: I agree, this is not yet optimal, but imo its not specific to ViewHelpers. Maybe we can call it `namespaceMapping` or `s.th.?`

(commenting from Symfony user group meeting ;)

#15 - 2012-03-08 08:57 - Adrian Föder

Hi Bastian,

| *the html tag is actually outside the rendered content.*

So it really means you have to declare the namespace to the "parent-most" template and it's inherited to all childs (i.e. partials and sections)?

Unfortunately this brings a major drawback: it makes Fluid not outputting the templates as-is anymore.

Currently, templates are output exactly as they are, only an occurrence of a {} or a foreign-namespaced tag is substituted by Fluid.

With that new <html> namespacing, this behavior is broken.

Additionally, as already said: the "local" namespacing won't be possible anymore, you'd have to decide a namespace token over the whole, complete project; and this is imho not the idea of a namespace (token).

The benefit of the tokens is, imo, the declaration in-place, that means: when examining code and coming across a namespace token, you know you find its "meaning" at the top of the file. So `c:` in one file would point to TYPO3.Comments, in the next file it may mean; whatever; Acme.Conversions.

I really **explicitly** loved exactly this circumstance; you must not be aware of not hurting existing namespaces or accidentally using a wrong one; being in one file makes you totally aware of on what NS you are on (with looking at the file header).

So, how's it done in plain XML? Unfortunately, when using plain XML you'd need a root node compulsorily. We would not want root nodes. So maybe it's possible to come back to the Element-based namespace declaration as discussed before,

```
<f:namespace
  xmlns:f="http://typo3.org/ns/2011/typo3/fluid"
  xmlns:cm="http://typo3.org/ns/2012/typo3/comments"
  xmlns:a="http://acme.org"
/>
```

(note it's an empty single tag).

the first xmlns:f it "required" to allow the f:namespace attribute at all; but I don't now if this is valid, really-parsable W3C conform syntax however.

As said, being correct would require a root node.

I don't have a real suggestion; another idea was to require an <f:template> root node in each file like it's usual in XSLTs...

But that's verbose and would break; but... that's not sooo bad, it could be reasonable with "each fluid file **must** be valid xml"... But again this reduces simplicity, imo....

#16 - 2012-03-08 09:48 - Sebastian Kurfuerst

Hey

| *So it really means you have to declare the namespace to the "parent-most" template and it's inherited to all childs (i.e. partials and sections)?*

If I understood Bastian correctly, that is not the case. If it were so, that'd be a no-go for me as well.

Greets, Sebastian

#17 - 2012-03-08 21:13 - Bastian Waidelich

Misunderstandings ;)

If you look at the patch (<https://review.typo3.org/#/c/9463/>) you can see that the xmlns syntax works in addition to the existing {namespace foo=bar} syntax with following exceptions:

- the **f** namespace identifier is skipped (because it is already registered by default)
- if a namespace can't be resolved (e.g. there is no mapping configured) the namespace will be skipped silently. Otherwise this would be a breaking change.
- the namespace declaration is **not** removed from the template

Namespaces will still be valid only for **one** Fluid file.

But you do have a valid point.. The files could look like this:

Layouts/Default.html

```
1<?xml version="1.0" encoding="utf-8" ?>
2<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
3<html xmlns="http://www.w3.org/1999/xhtml">
4  <head>
5    <title>Foo</title>
6  </head>
7  <body>
8    <f:render section="content" />
9  </body>
10</html>
```

Templates/Index.html

```
1<?xml version="1.0" encoding="utf-8" ?>
2<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
3<html xmlns="http://www.w3.org/1999/xhtml" xmlns:f="http://typo3.org/ns/typo3/fluid/viewhelpers" xmlns:vht="
http://typo3.org/ns/typo3/viewhelpertest/viewhelpers">
4
5  <f:layout name="Default" />
6  <f:section name="content">
7    <vht:hasAuto completion="true" />
8  </f:section>
9
10</html>
```

That works quite nice, but for partials that's not possible.. You can comment out the namespace declaration like:

Partials/Test.html

```
1<f:comment>
2  <html xmlns="http://www.w3.org/1999/xhtml" xmlns:f="http://typo3.org/ns/typo3/fluid/viewhelpers" xmlns:vht="
http://typo3.org/ns/typo3/viewhelpertest/viewhelpers">
3</f:comment>
4
5Partial content
```

But that's not very neat.. Maybe we can make this configurable so that the whole tag declaring namespaces is removed, but that might have unwanted side-effects and would be an overkill IMO.
Any other suggestions?

#18 - 2012-03-11 03:16 - Alexander Berl

If I may give my unqualified input on this matter, as I find the basic idea very interesting and promising (I always found it awkward to have the {namespace foo=bar} syntax):

I never really understood why I had to declare namespaces in partials anyway, since in my understanding their behaviour is completely different from

sections. As I see it, a partial is just that - a partial cutout of another bigger template, and as that internally it's just a copy&paste in place, so is just an extraction mainly for structuring/organization purpose, e.g.:

Templates/Index.html

```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xmlns:f="http://typo3.org/ns/typo3/fluid/viewhelpers"
xmlns:vht="http://typo3.org/ns/typo3/viewhelpertest/viewhelpers">

  <f:layout name="Default" />
  <f:section name="content">
    <f:render partial="MyPartial" />
  </f:section>

</html>
```

Partials/MyPartial.html

```
1 <vht:title>Hello World from the partial!</vht:title>
```

is basically the same as writing:

```
1 <?xml version="1.0" encoding="utf-8" ?>
2 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
3 <html xmlns="http://www.w3.org/1999/xhtml" xmlns:f="http://typo3.org/ns/typo3/fluid/viewhelpers"
xmlns:vht="http://typo3.org/ns/typo3/viewhelpertest/viewhelpers">
4
5   <f:layout name="Default" />
6   <f:section name="content">
7     <vht:title>Hello World from the partial!</vht:title>
8   </f:section>
9
10 </html>
```

hence the code in the partial ends up in the namespace of the template it lies within. However, Templates are completely different from that, as they get parsed separately and everything outside the `f:section` tags gets stripped away for rendering purpose. Hence IMO it is totally logical that they should be fully qualifying XML files and have their own namespace declaration, but not so much for partials. But that's maybe just my plain understanding of partials, which is not what you're thinking about.

The non-breaking solution I see would be to actually parse partials and remove the namespace declaring tag as Bastian suggested - but I'm not sure about that either, it would help in the IDE to have fully qualifying XML/XHTML files there too, but it's also awkward to have every partial wrapped inside a `<html>` tag.

Otherwise, why not wrap each partial inside a `<f:namespace>` tag as Adrian suggested and make it a passthrough `renderChildren` viewhelper (for it to be valid, the tag must really wrap the contents using the namespaces, though I'm not sure about whether using a namespace within the tag declaring it is valid - I'd blind guess not - on the other hand, you didn't need to declare the `f: namespace` until now either).

Would putting the `xmlns` declaration inside the `tag` be a plausible solution or would that be invalid syntax? I couldn't find a quick answer for that.

```
1 <?xml version="1.0" encoding="utf-8" xmlns:f="http://typo3.org/ns/typo3/fluid/viewhelpers"
xmlns:vht="http://typo3.org/ns/typo3/viewhelpertest/viewhelpers" ?>
2 Partial content
```

Then again, the `tag` would have to be stripped from each partial, too.

#19 - 2012-05-08 15:10 - Gerrit Code Review

Patch set 4 for branch **master** has been pushed to the review server.

It is available at <http://review.typo3.org/9463>

#20 - 2012-05-08 15:22 - Bastian Waidelich

Thanks for your comments and ideas.

We deliberately avoided some kind of namespace inheritance in Fluid, because that would make things much more complex & unpredictable for developers & integrators.

But I think, we found a good solution now:

If you use the default namespace pattern to generate your XSD files (<http://typo3.org/ns/Your/Package/ViewHelpers>) you won't have to configure any namespace mappings.

In order to use xml namespace declarations inside partials, you can use the "section" argument of the render ViewHelper like this:

SomePartial.html

```
1<html xmlns:x="http://typo3.org/ns/Your/Package/ViewHelpers">
2<body>
3<f:section name="body">
4  <x:someViewHelper />
5</f:section>
```

SomeTemplate:

```
1<f:render partial="SomePartial" section="body" />
```

#21 - 2012-05-08 15:59 - Gerrit Code Review

Patch set 5 for branch **master** has been pushed to the review server.

It is available at <http://review.typo3.org/9463>

#22 - 2012-05-08 16:01 - Gerrit Code Review

Patch set 6 for branch **master** has been pushed to the review server.

It is available at <http://review.typo3.org/9463>

#23 - 2012-05-08 16:06 - Gerrit Code Review

Patch set 7 for branch **master** has been pushed to the review server.

It is available at <http://review.typo3.org/9463>

#24 - 2012-05-08 16:21 - Gerrit Code Review

Patch set 8 for branch **master** has been pushed to the review server.

It is available at <http://review.typo3.org/9463>

#25 - 2012-05-08 17:53 - Bastian Waidelich

- *Status changed from Under Review to Resolved*
- *Target version set to 1.1 beta 1*
- *% Done changed from 0 to 100*