Fluid - ViewHelper - Incubator - Suggestion # 8779

Status:	Accepted	**Priority:**	Should have
Author:	Thomas Deinhamer	**Category:**	
Created:	2010-07-09	**Assigned To:**	
Updated:	2012-08-29	**Due date:**	
Has patch:			
Tags:			
Subject:	format.filesize		

Description

Attached is a basic filesize formatting viewhelper for TYPO3 v4.
It's still missing some functions like the ability to disable
the unit suffix. Also localization features are missing, e.g.
defining the symbol which should be used as comma or thousands separator.

Feedback is highly appreciated!

History

#1 - 2010-07-10 13:27 - Bastian Waidelich

- Status changed from New to Accepted

Looks good!

If following "issues" would be resolved, we could add this VH to the Incubator distribution:

1. Class is **Tx_Hype_ViewHelpers_Format_FilesizeViewHelper** should be **Tx_ViewhelperIncubator_ViewHelpers_Format_FilesizeViewHelper**
2. A v5 version is missing (just replace **Tx_..._...** by **F3\...\...**
3. Unit tests are missing

Furthermore one or two examples in the source code (above the class declaration) would be good. Like:

```
* [...]
* = Examples =
*
*
* <f:format.filesize size="123456" />
*
*
* Output:
* xyz
*
* = Examples =
*
*
* {f:format.filesize(size: 999999, unit: 'm', precision: 3, mode: 'decimal')}
*
*
* Output:
* xyz
```

And last but not least some nitpicks that wouldn't prevent the VH to be included IMO, but should be considered (I hope, you don't mind nitpicking):

- To be more meaningful to the user, the parameter **size** could be renamed to s.th. like **bytes** or **sizeInBytes**
- String-parameters with a fixed set of allowed values are not easy to comprehend and sometimes error prone. Instead of the **mode** parameter, you could have a boolean **decimal** or **isDecimal** parameter. If there will be more modes sometime, you can still refactor this, but I guess thats a case of YAGNI ;)
- its not easy to understand, what $this->modes[$mode] is at the first glance. With the above suggestion it might be rewritten to $quotient = $isDecimal ? 1000 : 1024;

Apart from that, it's a very neat and useful ViewHelper, thanks for sharing!

#2 - 2010-07-10 13:29 - Bastian Waidelich

BTW: This shouldn't be read as some kind of todo list or request. I was justs collecting some tasks that should be solved by **someone** in order to get this VH into the Fluid core.

#3 - 2010-07-10 17:14 - Thomas Deinhamer
- File FilesizeViewHelper.php added

Added a new version with some changes and new features.
Now it's possible to define a decimal and a thousands separator.

Bastian, some of your suggestions found their way into the new version.
Regarding the suggestion to change the 'size' attribute I am not sure
if we should do that, because 'size' is rather short and I think it's
describing what to provide, very well. Maybe we can find another phrase
for it, which is still short but matches better.

Added 3 different usage examples. Now it's possible to choose
from 3 different syntaxes, 2 of them beeing tag-based.

I renamed the 'precision' attribute to 'decimals' to be in sync
with other viewhelpers, which are already in Fluid. Also the 2
new attributes 'decimalSeparator' and 'thousandsSeparator' are
named exactly the same like the ones in the Fluid number view helper.

Please add more feedback, I'm also for getting a polished version
of the filesize viewhelper into the package or even into Fluid.

#4 - 2010-07-12 13:12 - Bastian Waidelich
- Subject changed from Add filesize formatting. to format.filesize

#5 - 2010-09-13 15:20 - Christian Zenker
- File ByteViewHelper.php added

How about renaming the VH to format.byte? The VH does not depend on a file in any way - this way the naming is more suitable.

After all I'd like to have some "auto-detection" of the best suitable output-format. Consider having files with high variety sizes (like 5kB to 100Mb) - you won't find any parameters for the VH that allows rendering in a nice way. But this would make it much simpler to use. We could either entangle this functionality in the current version or create a new one. Benefit of the second version would be easier handling in templates (<f:render.byteAuto size="123567890" /> looks better than <f:render.byte size="123567890" auto="1" />).

I append a viewHelper I wrote for that purpose. Its a bit customized and not configurable yet but might be a good example on how the viewHelper could look like.

#6 - 2010-09-13 15:55 - Thomas Deinhamer

Hello!

The idead about renaming the viewhelper to something other than 'filesize'
sounds good - but I think it needs some further discussion.

The "automatic" feature you're talking about, is already implemented in the
viewhelper I provided - but I'd be glad if you could provide some further
information about your code. E.g. what's up with the number 820?

#7 - 2010-09-13 16:09 - Christian Zenker

> *The "automatic" feature you're talking about, is already implemented in the*
> *viewhelper I provided*

Well, I didn't pay attention to that. ;)

> *but I'd be glad if you could provide some further*
> *information about your code. E.g. what's up with the number 820?*

These are 0.8kB. That's the customisation I was writing about. I'd like to display 0.9kB rather than 982B as it is catchier. That's why I used this factor. But I don't think it had to be implemented in a standard VH (although it might).

#8 - 2010-09-13 16:14 - Thomas Deinhamer

Christian Zenker wrote:

> > *The "automatic" feature you're talking about, is already implemented in the*
> > *viewhelper I provided*
>
> *Well, I didn't pay attention to that. ;)*
>
> > *but I'd be glad if you could provide some further*
> > *information about your code. E.g. what's up with the number 820?*
>
> *These are 0.8kB. That's the customisation I was writing about. I'd like to display 0.9kB rather than 982B as it is catchier. That's why I used this factor. But I don't think it had to be implemented in a standard VH (although it might).*

Ah, got it now. That's imo a nice feature, I've seen that somewhere else too.
It's just a matter of giving this variable a proper name and use it in the

calculation loop inside the viewhelper instead of the 1024/1000 bytes.
Maybe I'll come up with an updated version today evening if I find the time.

#9 - 2010-09-13 23:15 - Thomas Deinhamer

- File FilesizeViewHelper.php added

Attached is an updated version. It has the new feature Christian proposed;
now you can define a calculation threshold. If the final size is greater than
the threshold, the next unit is chosen and the value is recalculated.
Also a little bug was fixed in this version where an exact filesize
of 1024 Bytes would still stay 1024 B, instead of converting to 1 KiB.
(And even 1024 KiB would stay, not converting to 1 MiB, and so on.)
What's still missing is a check if we run out of the SI units.
Have fun!

#10 - 2010-09-13 23:22 - Thomas Deinhamer

There are still some things to fix/consider.
Refering to wikipedia (http://de.wikipedia.org/wiki/Byte)
the unit for kilobyte is written "kB" and not "KB", while
all the others are uppercase (MB, GB, etc.). The binary equivalents
are all formatted identically: KiB, MiB, etc. Sure more to come.

#11 - 2011-02-22 16:10 - Thomas Deinhamer

- File FilesizeViewHelper.php added

Attached a new version which corrects some bugs when the threshold was set to 0 and it also resets the units array for every call, to calculate a correct filesize. Also removed unneeded whitespace.

#12 - 2012-08-29 16:44 - Giulio Iotti

Line 96 should be:

```
    $unit = strtoupper($unit) . ((!$useDecimalSystem && $unit != 'b') ? 'i' : '') . ($unit != 'b' ? 'B' : '');
```

Files

FilesizeViewHelper.php	2.9 kB	2010-07-09	Thomas Deinhamer
FilesizeViewHelper.php	3.7 kB	2010-07-10	Thomas Deinhamer
ByteViewHelper.php	2.3 kB	2010-09-13	Christian Zenker
FilesizeViewHelper.php	4.3 kB	2010-09-13	Thomas Deinhamer
FilesizeViewHelper.php	4.2 kB	2011-02-22	Thomas Deinhamer