

## EXT: Fluid-based Display

Extension Key: datadisplay

Language: en

Keywords: forEditors, forIntermediates

Copyright 2011, Francois Suter (Cobweb), <typo3@cobweb.ch>

This document is published under the Open Content License  
available from <http://www.opencontent.org/opl.shtml>

The content of this document is related to TYPO3  
- a GNU/GPL CMS/Framework available from [www.typo3.org](http://www.typo3.org)

## Table of Contents

EXT: Fluid-based Display.....	1	The sample Fluid template.....	5
<b>Introduction.....</b>	<b>3</b>	Using Fluid partials.....	6
What does it do?.....	3	Localized strings.....	6
Credits.....	3	<b>Developer's guide.....</b>	<b>7</b>
Questions?.....	3	Hooks.....	7
Keeping the developer happy.....	3	View helpers.....	7
<b>Installation.....</b>	<b>4</b>	<b>Known problems.....</b>	<b>8</b>
<b>User's manual.....</b>	<b>5</b>	<b>To-Do list.....</b>	<b>9</b>
Input screen.....	5	<b>ChangeLog.....</b>	<b>10</b>

# Introduction

## What does it do?

This extension provides a Data Consumer (see extension “tesseract” for a general explanation) which relies on the Fluid template engine for rendering data provided by Tesseract Data Providers. This makes it a very flexible tool.

## Credits

This extension was developed by Fabien Udriot (Ecodev).

It takes advantage of all the power of Fluid, so credits also go to Sebastian Kurfürst and all others who contributed to developing and enhancing Fluid.

## Questions?

If you have any questions about this extension, you may want to refer to the Tesseract Project web site (<http://www.typo3-tesseract.com/>) for support and tutorials. You may also ask questions in the TYPO3 English mailing list ([typo3.english](mailto:typo3.english)).

## Keeping the developer happy

If you like this extension, do not hesitate to rate it. Go the Extension Repository, search for this extension, click on its title to go to the details view, then click on the “Ratings” tab and vote (you need to be logged in). Every new vote keeps the developer ticking. So just do it!

You may also take a step back and reflect about the beauty of sharing. Think about how much you are benefiting and how much yourself is giving back to the community.

## Installation

Install "fluiddisplay" as part of the Tesseract suite. It cannot be used alone.

Obviously "fluiddisplay" depends on "fluid". It also requires **TYP03 4.5 or more** to run.

# User's manual

There's nothing to configure in order to use Fluid Display. All that's needed is to define a Fluid template in a file and create a relation to that file.

## Input screen

This is the Fluid Display input screen. Apart from the title, the only really important parameter is the reference to the Fluid template file.

**Hide:**  
☐

**Title**  
All content elements

**Description**

**Fluid Template**  
EXT:fluiddisplay/Samples/sampleTemplate.html

The **Hide** is not currently used, although it could be in the future. In the meantime it can still be used to indicate an unused filter.

The template file can be located either in the "fileadmin" hierarchy or in an extension (using the "EXT:" syntax).

## The sample Fluid template

A sample template file is provided in the "Samples" folder of extension "fluiddisplay". It should give you a pretty good idea of what can be achieved and how to manipulate the Data Structure with Fluid. The sample template assumes that the following Data Query has been executed:

**General** **Advanced**

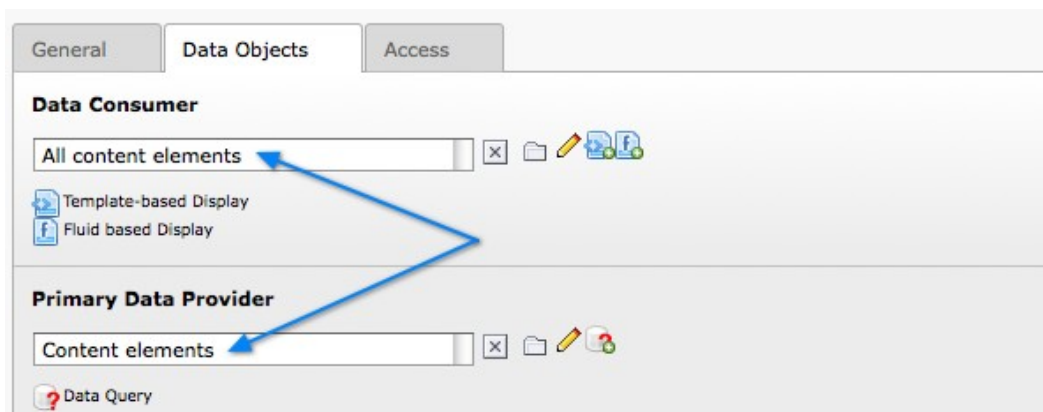
**Hide:**  
☐

**Title**  
Content elements

**Description**

**SQL Query**  
SELECT \* FROM tt\_content ORDER BY header

The elements can simply be related with a standard Display Controller:



## Using Fluid partials

It is possible to use Fluid partials inside the template, as is demonstrated in the sample template provided by the extension. The rule to follow is simple: the partials must be stored in a "Partials" folder located at the same place as the same template.

### Example:

Assuming your template file is located in:

```
fileadmin/templates/fluid/myTemplate.html
```

Fluid will expect your partials to be in:

```
fileadmin/templates/fluid/Partials/
```

## Localized strings

It's not possible to have a pre-loaded locallang file in the case of "fluiddisplay". So the full "LLL:" syntax must be used in each translate view helper.

# Developer's guide

This chapter contains information targeted at developers.

## Hooks

Fluid Display contains one hook:

- **preProcessView**: this method will receive two parameters. The first one is a reference to the Fluid view object that will be used for rendering (an instance of `Tx_Fluid_View_StandaloneView`). The second parameter is a back-reference the current instance of `tx_fluiddisplay`. This hook mostly makes it possible to assign additional variables to the Fluid view.

## View helpers

There are not many hooks in Fluid Display mostly because it relies on Fluid for rendering, which is already powerful and flexible enough. If you have some special needs in your rendering the way to go is to create custom view helpers.

Some information about writing view helpers can be found in the Fluid documentation:

<http://flow3.typo3.org/documentation/manuals/fluid/fluid.usermanual/#fluid.usermanual.writingaviewhelper>

Looking at existing view helpers is also very useful.

## Known problems

If you have any issues, please refer to the Tesseract Project web site (<http://www.typo3-tesseract.com/>). You may also post your problems to the TYPO3 English mailing list ([typo3.english](mailto:typo3.english)), so that others may benefit from the answers too. For bugs or feature requests, please open an entry in the extension's bug tracker on Forge (<http://forge.typo3.org/projects/extension-fluiddisplay/issues>).



## To-Do list

The roadmap for the evolution of this extension can be found on Forge: <http://forge.typo3.org/projects/roadmap/extension-fluiddisplay>

For feature requests, please open a report on Forge issue tracker: <http://forge.typo3.org/projects/extension-fluiddisplay/issues>

## ChangeLog

Version	Changes:
1.0.0	First public version