

## TYPO3 Core - Task #83776

Epic # 84101 (Closed): The core must provide configurable structures for both pages and elements out of the box

### Definition of goals to create the stories

2018-02-05 12:46 - Jo Hasenau

<b>Status:</b> Closed	<b>Start date:</b> 2018-02-05
<b>Priority:</b> Must have	<b>Due date:</b> 2018-02-07
<b>Assignee:</b>	<b>% Done:</b> 0%
<b>Category:</b> Backend User Interface	<b>Estimated time:</b> 0.00 hour
<b>Target version:</b> Candidate for patchlevel	<b>Complexity:</b>
<b>TYPO3 Version:</b> 9	<b>Sprint Focus:</b>
<b>PHP Version:</b> 7.2	
<b>Tags:</b>	
<b>Description</b> Everybody who wants structured content to become part of the core, please write down the reasons why you think this should happen. Just write some short sentences so everybody will quickly get your reasoning.  To make sure we are talking about the same things, please read the discussion here first: <a href="https://wiki.typo3.org/Talk:Blueprints/ContentElementDefinitions">https://wiki.typo3.org/Talk:Blueprints/ContentElementDefinitions</a>  We will then discuss these reasons on Slack <a href="https://typo3.slack.com/messages/C8Z2UM50Q/">https://typo3.slack.com/messages/C8Z2UM50Q/</a> and generate user stories based on these reasons. So before deciding HOW we are going to do it, everybody will know WHY we want it and WHAT we are going to achieve.	

### History

#### #1 - 2018-02-05 13:22 - Markus Klein

use case 1:

I need to have a good visual representation of multi-column content in BE. example: putting to arbitrary content elements next to each other

use case 2:

I need to collect multiple content elements, which can be shown in a carousel.

use case 3:

I need to logically group content elements together. example: warp those CEs with certain markup in FE output.

#### #2 - 2018-02-05 13:30 - Jo Hasenau

- Status changed from New to Needs Feedback

1. I want to enable editors to make use of upcoming features like the CSS Grid Layout Module <https://www.w3.org/TR/css-grid-1/>
2. I want to speed up daily work by grouping single elements so people can apply actions like "move" or "copy" to just one container instead of each single element.
3. I want the backend to be working without static structures, so people can use it on mobile devices too.

#### #3 - 2018-02-05 14:32 - Riccardo De Contardi

1. I need to create nested containers of content elements
2. I need to create structures like tabs and accordions; these kind of structures (especially the first one) may require to pass some information or content from the content inside the container to the container itself. - may be linked to use case 3 by Markus Klein

#### what I don't need

1. As integrator *I don't need to generate bloated frontend code* that is difficult to be adapted to my needs; I would want to be in full control of the output
2. As editor *I don't need to be overwhelmed with so many configuration options* that I can't understand even half of them
3. As integrator *I don't need to be forced to use Bootstrap grid* (if I prefer Susy or Foundation), so if you want to consider a feature that builds the frontend grid based on the backend layout, please consider that if you stick to Bootstrap you will force people that wants to use a different framework to do extra labor.

Generating a CSS with display:grid would be awesome, but we need to consider the compatibility with IE 10 and 11

#### #4 - 2018-02-05 16:28 - Tobias Wollender

I'm currently avoiding something like gridelements completely - just using "components" tailored to the customer's needs. So for "multicolumn" teasers (or accordions, tabs, sliders etc.) I use special elements (components) with child-elements (through IRRE). I can then define the backend fields and frontend output exactly like I (or the customer) want it. So at the moment I wouldn't need "structured content elements" (if you say that this is something like gridelements). I would prefer an (enhanced) solution like mask in the core, to build / edit custom components more quickly and reliably.

#### #5 - 2018-02-06 10:07 - Annett Jähnichen

Here is our summary for Use Cases:

1. I need a visual structure of multiple columns with arbitrary content both in frontend and in backend with a minimum amount of page templates (flux) / backend layouts
2. I need a possibility to move nested content elements from one column or row to another via drag & drop
3. I need a flexible amount of rows in a structure (e.g. an accordion) that can be easily extended (plus button) or decimated (trash button)
4. I need a flexible amount of rows and columns where the column widths can differ from row to row as one structure (e.g. a matrix or mosaic). Without having to create it with custom content elements (something like: container 70:30, container 30:70, container 70:30 ...)
5. I need to be able to reorder the rows of a structure (e.g. an accordion) via drag & drop
6. I need a manually added (not generated by e.g. backend layouts) backend section which represents a frontend section (<section>) that can contain other structured content elements.
7. A parent content element needs to know its children.
8. A child content element needs to know its parent.
9. I need nested content elements to be properly translated with its parent

#### #6 - 2018-02-06 10:18 - Felix Nagel

Annett's stories are pretty good, though I would like to add another point:

I want to have an option to add flexform like configuration to columns (e.g. for adding different width for breakpoints, force equal height, etc.)

#### #7 - 2018-02-06 14:56 - Moritz Ahl

- *File add and change rows.png added*

- *File edit properties.png added*

- *File inline-editing-content.png added*

One remark: Let's not re-invent the wheel UI-wise. Have a look on wordpress page builder's UI (e.g. visual composer). I don't like them as a developer, but I love them as an editor! See the screenshots attached.

Some other thoughts:

- I think a "custom content" element (slider, accordion) is another thing as a "container" element which provides layout structure (row, column). It needs a different UI, behaves differently and the page module is the place to interact with it. And is the relation between a column and its content the same thing as between a slider and its slide? While it might be technically alike, it is a different thing UI wise. The implementation should consider this.
- Do we really need infinite nesting of rows/columns into other rows/columns, for example 50:50 inside another 50:50? Often it's just a laziness thing because it takes so long to create 4 columns à 25.
- maybe it is wise to limit the scope to handling rows and columns - and do this one thing well. There are solutions for custom elements out there which would still integrate nicely.

My main arguments to put this into the core:

- Editing layout/structure (read: rows and columns) should happen in the **page module**. Re-Arranging cols/rows, inline editing and drag n drop play an important role to make it easy to use for editors. And since the page module is part of the core and hard to extend or modify, I think the core should address this topic.

Here's some more stories:

As an editor...

- I want to create rows "on the fly" inside a page or above or below other rows, in order to place content or columns in it.
- I want to be able to insert content or columns into rows to place content in it.
- I want to create multiple columns in one row with predefined divisions (for example, 1/3 + 2/3) with just two clicks
- I want the visual representation of the columns to reflect their widths.
- I want the last column exceeding the max. available row width to break up to the next "line"
- I want to move rows below or above other rows (nowhere else!)
- I want to move columns before or next to other columns (nowhere else!). It must work even across rows.
- when moving rows or columns, I want their columns and content to move with them.
- I want to delete and duplicate rows, columns or content elements.
- I want to edit the properties of rows and columns in a modal to change behavior or looks quickly, for example the width.
- I want all changes to the structure (create, delete, move, edit properties) to be visible instantly, without reload or leaving context.
- I want to quickly change the divisions of all columns in one row preserving the content in it
- I want to cut, copy and paste rows (including their content) across pages.

As a developer...

- I want to be able to define a grid system in one template instead of independent files for each configuration (30-70, 70-30, 20-80, 20-60-20, ...)
- I want to re-use grid elements across projects and be able to download for example bootstrap grid definitions with pre-defined properties for columns and rows which make sense for almost any project (width as in 1 to 12, responsive behavior settings, maybe padding and margin?) with a nice UI
- I want to be able to define custom properties for rows and columns as I like

#### #8 - 2018-02-06 16:15 - Lidia Demin

In addition to [#83776#note-5](#)

As a developer:

- I want to be able to change class names (override template files) for grid structures (similar to what EXT:form does). Is important for custom designs with custom HTML templates
- I want to be able to restrict the editor in some actions to avoid him breaking a less flexible website design

#### #9 - 2018-02-06 17:10 - Wiebke Lauke

This would be really nice but it's a step further into the future:

- there could be a migration tool to migrate gridelements into core structured CEs

When you think of TemplaVoila content elements (fce) there's no great possibility to migrate.

#### #10 - 2018-02-06 22:12 - Andreas Steiger

- File *wix-ui.gif* added

- File *shopware-ui.gif* added

I really like the thoughts of Moritz Ahl. We should orient ourselves to well-functioning concepts. But not just Wordpress ;) See my short screencasts attached. We also can have a look on website builder WIX (I know this is frontend editing, but I like the UX for creating columns) and the new Emotion/landingpage Builder of Shopware. Of course, this builder for landingpages and other stuff is a bit buggy and when I set a 12-column layout, it does not work anymore on mobile screens... but I love the concept of switching viewports and linking elements between the different viewports in a simple way. This is an innovative UI for managing content for responsive screens I think.

My additional use cases and stories:

1. As an editor, I want to split quickly my content into 2,3 or 4 columns with an user-friendly UI
2. As an editor, I want to put my content in a section/row in order to separate it visually in the frontend. (Alternating colored backgrounds on a landingpage)
3. As an editor I don't want to put a container item in a container item within a container item
4. As an editor I want to reorder my content for mobile devices and hide elements
5. As an editor I want to synchronize my content between viewports, but also be free to cut the synchronization for one special viewport
6. I need a multi-selection of content elements to move more than one item (without additional container, as it is with EXT:gridelements)
7. I don't need a confusing wireframe view with many many nested containers and a horizontal scrollable Screen
8. I need strong defaults and a editor-friendly UI (not developer-friendly)

#### #11 - 2018-02-07 05:29 - Falk Röder

I think using flexbox instead of floating is the more modern way for laying out things and often it makes more sense, to go for a content driven design/layout. But using the full power of this:

- I may need to create a flexible page layout depending on content I put on the page.
- For that I need to configure flexbox options for each column, row and content element. And to make it complete (and unfortunately really complex): all that with different media queries.

By following this approach 100%:

- it would be needed to create CSS Stylesheets on the fly by considering the configuration (containing media queries and styles for each element).

#### #12 - 2018-02-07 09:22 - Andreas Steiger

The most modern and innovative way is certainly CSS Grid ;) e.g. <https://gridbyexample.com/examples/example12/>  
Because we need a grid base an not just only a row base

#### #13 - 2018-02-07 11:06 - Alexander Opitz

- I want content elements which will be on all pages down the tree
- I want content elements on multiple pages as reference (so only one place to edit them)
- I want different order/show/hides of content elements depending on medium (mobile/tablet/desktop/TV)
- I want to define which CE can be put in which col/rowtype/container

**#14 - 2018-02-07 11:17 - Jigal van Hemert**

Some non-structured thoughts (usage of must, should, may, etcetera is not strict):

- some sites require a flexible use of multiple columns (or maybe even more complex layouts) with content
- current flexform implementation has quite a few restrictions (FormEngine has restrictions with evaluation of displayCond for example) and the storage method in the database is not to everyone's liking
- structured content elements also need some extra fields besides areas with content elements
- translation of structured content elements and the content elements they contain must be taking into consideration
- when changing the type of a structured content element, content elements may become orphans (e.g. changing from a three column to a two column block). (The page module already shows this for content elements belonging to an unused colpos). Some mechanism to warn editors and let them solve the issue must be made
- integrators would like it if there was an easy mechanism to create structured content elements, similar to backend layouts
- the definition of structured content elements must be stored in such a way that it can be used in a versioning system
- integrators should be able to restrict the type of content that can be used in content areas
- frontend rendering of structured content elements should be done with Fluid templates. There should be a mechanism to render the content areas

**#15 - 2018-02-07 11:43 - Alexander Opitz**

And please, no speculation about migration from gridelements/flux/dce/TV+ ... thats a part after the core work.

**#16 - 2018-02-07 12:40 - Rachel Foucard**

Dear Santa Claus,

as an editor, I'd be really happy:

- if I could change, delete or add columns in a page without creating multiple nested contents.
- if I could manage the columns of a page like with the be\_layout wizard
- if the columns of a page could be inherited automatically for child pages
- if I could easily drag and drop my content in these columns
- if these columns could automatically have responsive behavior and if I could adjust this behavior
- if changing the columns of a page could be well managed in workspaces
- if I could easily drag and drop unused content in these columns
- if the specific settings for displaying a page (checkbox, drop-down list) were directly accessible without opening the complete page form.

as a developer, I'd be super happy:

- if the columns of a page were no longer defined by nested contents (but by be\_layout)
- if the new design of a site no longer required me to make complicated scripts to replace the contents of an old position to a new position.
- if the columns of a page could be easily set up to match the bootstrap behavior.

If the structured content project is about something other than making columns freely, I am currently very happy with Mask (for specific content), and I find that bootstrap\_grid (with gridelement) for columns is an acceptable (UX) solution.

is it possible to contribute ideas in terms of UX?

Cheers,  
Rachel

**#17 - 2018-02-07 13:11 - Coders.Care Extension Team**

[Rachel Tjong-A-Hung](#) of course everybody is invited to contribute ideas in terms of UX, since this is the original motivation of the common interest group: To improve the TYPO3 user experience for each target group from editors to integrators to developers.

**#18 - 2018-02-07 16:55 - Rachel Foucard**

- File *be\_layout-management-ideas.jpg* added

Cool!

Here's a picture of what I'm thinking.

Cheers,  
Rachel

**#19 - 2018-02-08 10:37 - Alexander Opitz**

- I want the possibility of A/B- and Multivarianttests

**#20 - 2018-02-10 18:16 - Riccardo De Contardi**

[Rachel Tjong-A-Hung](#) Foucard your ideas seem good; Just some non-structured thoughts

- also interaction via mobile devices should be considered
- I should be able to "lock" the addition or subtraction of columns and rows. I explain myself better: the following scenarios should be considered

- a) I have already structured my page into two (or more) columns (like now) and I just want to add content in them without sub-splitting.
    - a.1) I don't know if the adding of a new CE could be assimilated to the adding of a row
    - b) I want to provide some already configured row structures (2-cols, 3-cols) without letting people modify them (maybe the proportions)
- So we have at least 3 possible scenarios (in order of flexibility low -> high)

- 1) Just add content element, the structure is already provided
- 2) add structures that are already built
- 3) build the structure (rows, columns and their proportions) and then add the contents (your solution)

- I should be able to alter some "settings" of each row: for example I would like to set a different background for each row. So each row could have a "settings" button

**#21 - 2018-03-01 12:30 - Jo Hasenau**

- Parent task changed from #67134 to #84101

**#22 - 2018-09-06 14:20 - Susanne Moog**

- Target version changed from 9 LTS to Candidate for patchlevel

**#23 - 2020-03-12 11:27 - Georg Ringer**

- Status changed from Needs Feedback to Closed

**Files**

---

add and change rows.png	81.4 KB	2018-02-06	Moritz Ahl
edit properties.png	121 KB	2018-02-06	Moritz Ahl
inline-editing-content.png	109 KB	2018-02-06	Moritz Ahl
wix-ui.gif	13 MB	2018-02-06	Andreas Steiger
shopware-ui.gif	20 MB	2018-02-06	Andreas Steiger
be_layout-management-ideas.jpg	1.76 MB	2018-02-07	Rachel Foucard