A LIST APART

Illustration by Kevin Cornell

Frameworks

by Ethan Marcotte · November 24, 2015

Published in Responsive Design

A note from the editors: We're pleased to share an excerpt from Chapter 5 of Ethan Marcotte's new book, *Responsive Design: Patterns & Principles*, available now from A Book Apart *(http://abookapart.com/products/responsive-design-patterns-principles)*.

Over the past few years, we've been learning how to adapt our layouts to the infinite canvas of the web. Our sites can be viewed on any size screen, at any time, and responsive design is one approach that lets us accommodate the web's variable shape. But with all of the challenges we're facing and those yet to come, we need to begin building not just patterns, but *principles* for responsive design—principles that will allow us to focus not just on layout, but on the quality of our work.

If each part of your responsive interface is more or less self-contained—with its own layout rules, content needs, and breakpoints—then the code behind each element's design is far less important than thinking carefully about *how* and *why* an element should adapt. In other words, how do we move beyond thinking in terms of columns and rows, and start talking about the quality of our responsive designs? And what would frameworks to support that look like?

Finding the words

Honestly, there's no perfect answer to that question. But recently, a number of designers and organizations have started sharing the vocabulary they use to decide how and when their responsive designs should adapt. Vox Media, for example, thinks of their content as existing within a river—and in keeping with the metaphor, the flow of that content can be interrupted at certain points. Here's how they describe the front pages of Vox.com *(http://bkaprt.com/rdpp/05-16/)*:

> *Content flows around "rocks" and "breakers," which are modules such as a "Most Commented" list or a row of "Popular Videos." Many of these behaviors remain in the new layout system, but the key difference is an added contextual layer. Elements in the river are laid out to better highlight the diversity of content on Vox—articles, features, videos, editorial apps, card stacks, to name a few. Each one displays differently depending on its type and neighboring entries.*

Note that the language they use to talk about the quality of their layouts doesn't revolve around columns or rows. There's nary a mention of grids. For Vox, the design process begins with content priority and evolves into a layout. By understanding the weight and importance of each piece of content that flows through the river, the Vox team can algorithmically generate a responsive layout that best reflects the importance of the information within it.

Starting with an abstract system of columns and rows would be wrong for them—and, I'd argue, for every web designer. After all, according to Mark Boulton *(http://bkaprt.com/rdpp/05-17/)*, there are three fundamental benefits of a grid system:

- Grid systems **create connectedness**. A well-made grid can visually connect related pieces of content or, just as importantly, separate unrelated elements from each other. In other words, they help us create narratives from our layout.
- By establishing predefined alignment points, grid systems help designers **solve layout problems**.
- A well-designed grid system will **provide visual pathways for the reader's eye to follow**, and allow them to better understand a visual hierarchy.

As Boulton notes, we historically created grid systems by adopting a "canvas in" method. Working from the edges of a printed page, designers would subdivide a page into a system of columns and rows, and place images and text upon that grid in pleasing, rational arrangements. But the web doesn't have any such boundary—after all, it's the first truly fluid design medium. As a result, Boulton argues we should instead adopt a "content out" approach to designing our grids: to build more complex layout systems out from a foundation of small, modular pieces of content. And to do so, Boulton proposes three guiding principles:

- **Define relationships from your content.** A grid for the web should be defined by the content, not the edge of an imaginary page.
- **Use ratios or relational measurements above fixed measurements.**
- **Bind the content to the device.** Use CSS media queries, and techniques such as responsive web design, to create layouts that respond to the viewport.

By understanding the shape of our content, we can create flexible layouts that support connectedness—not just between related pieces of information, but between our layouts and the device. We can make responsive grid systems that don't just fit on an ever-increasing number of screens—they'll feel at home, wherever they're viewed.

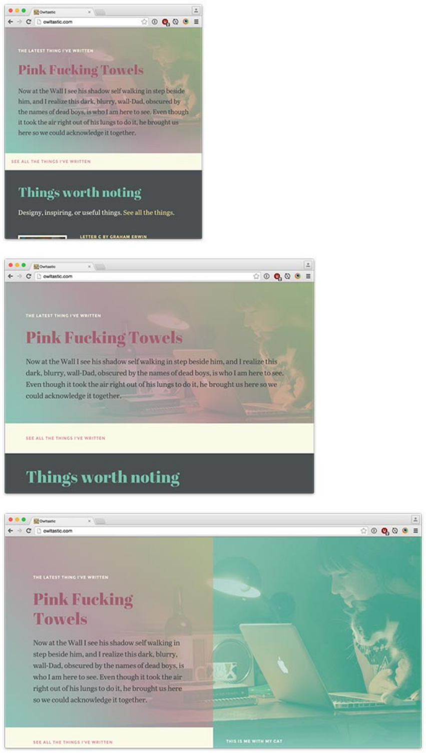
Finding the seams

Principles are wonderful, of course, but we still have to find a means of implementing them: of translating those ideals into practical responsive patterns and layouts. For me, that "content out" process begins by looking at the smallest version of a piece of content, then expanding that element until its seams begin to show and it starts to lose its shape. Once that happens, that's an opportunity to make a change—to introduce a breakpoint that reshapes the element and preserves its integrity.

But first, we need a method of finding an element's seams, and understanding how it loses its shape. For me, that process begins by looking at four characteristics: **width**, **hierarchy**, **interaction**, and **density**.

WIDTH

Width might be a little self-evident. As the width of a viewport changes, so does the width of a responsive design. But as the design gets wider or narrower, so will the elements within it, and as those modules expand or contract, there may be opportunities to add a breakpoint.



Pink Fucking Towels

Now at the Wall I see his shadow self walking in step beside him, and I realize this dark, blurry, wall-Dad, obscured by the names of dead boys, is who I am here to see. Even though it took the air right out of his lungs to do it, he brought us here so we could acknowledge it together.

Things worth noting

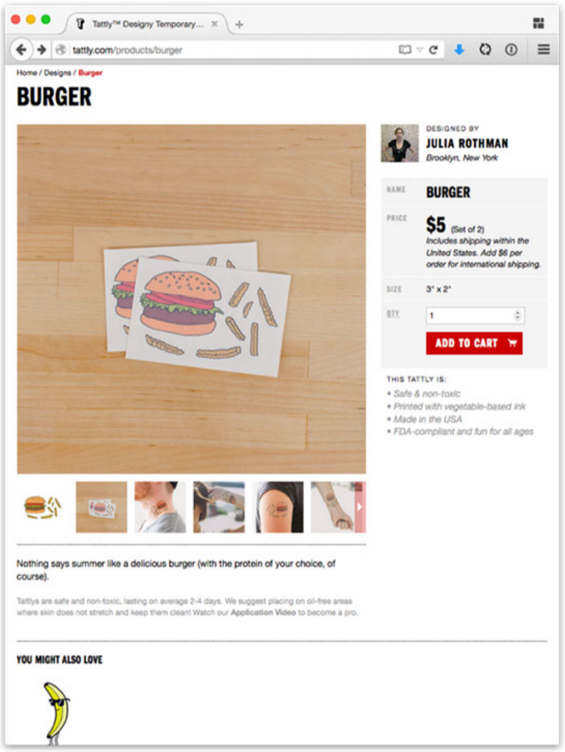
Designy, inspiring, or useful things. See all the things.

On her stunning responsive portfolio, Meagan Fisher adjusts the typography of certain elements—not just their layout—as their width expands and contracts.

HIERARCHY

Width is, I'm sure you'll agree, the most common characteristic of a responsive design—but it's not the only one. As the shape of an element changes, the *hierarchy* of elements may need to change as well.

Let's take a quick look at a product page on Tattly's responsive ecommerce site *(http://bkaprt.com/rdpp/05-19/)*. When viewed on wider screens, the primary content area has two key pieces of information: a photo carousel of the product on the left, and a call to action to purchase the product on the right.



BURGER

DESIGNED BY
JULIA ROTHMAN
Brooklyn, New York

NAME	BURGER
PRICE	$5 (Set of 2) Includes shipping within the United States. Add $6 per order for international shipping.
SIZE	3" x 2"
QTY	1

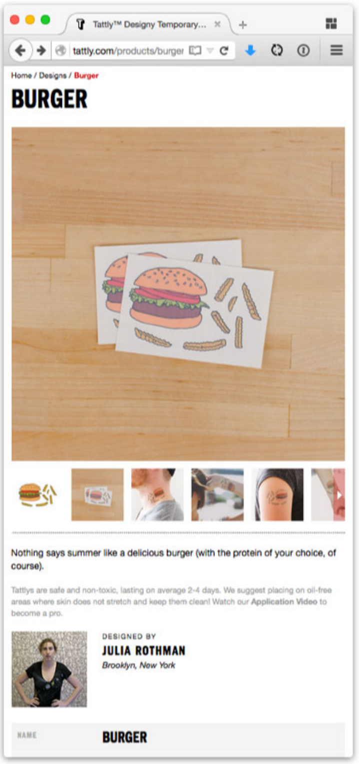
THIS TATTLY IS:
- Safe & non-toxic
- Printed with vegetable-based ink
- Made in the USA
- FDA-compliant and fun for all ages

Nothing says summer like a delicious burger (with the protein of your choice, of course).

Tattlys are safe and non-toxic, lasting on average 2-4 days. We suggest placing on oil-free areas where skin does not stretch and keep them clean! Watch our **Application Video** to become a pro.

On Tattly's responsive ecommerce site, the product content is laid out in a pleasing two-column grid on wider screens.

But that's just one view of this particular part of the design, because as screens get narrower, we lose the ability to place multiple columns side by side. That's where a question of hierarchy arises: in a single-column layout, which piece of content should appear first? Tattly opted, quite rightly, to lead with photos of the product—but you may answer hierarchy questions differently on your site.

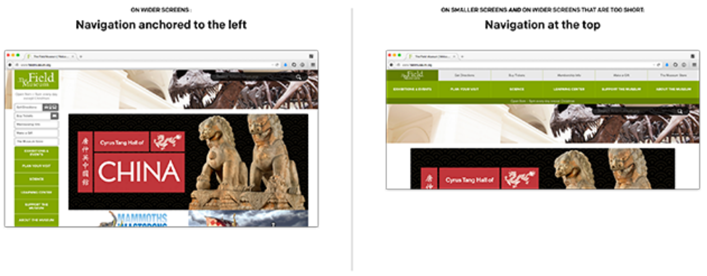


Nothing says summer like a delicious burger (with the protein of your choice, of course).

Tattlys are safe and non-toxic, lasting on average 2-4 days. We suggest placing on oil-free areas where skin does not stretch and keep them clean! Watch our Application Video to become a pro.

On narrower viewports, the hierarchy of product information shifts from two columns to one.

Hierarchy is generally a reminder to be more *vertically aware* in our designs. After all, we have min-width and max-width media queries, but can also avail ourselves of min-height and max-height queries more often. I think the navigation menu for the Field Museum *(http://bkaprt.com/rdpp/05-20/)* beautifully balances vertical and horizontal layouts.


ON WIDER SCREENS:
Navigation anchored to the left
ON SMALLER SCREENS AND ON WIDER SCREENS THAT ARE TOO SHORT:
Navigation at the top

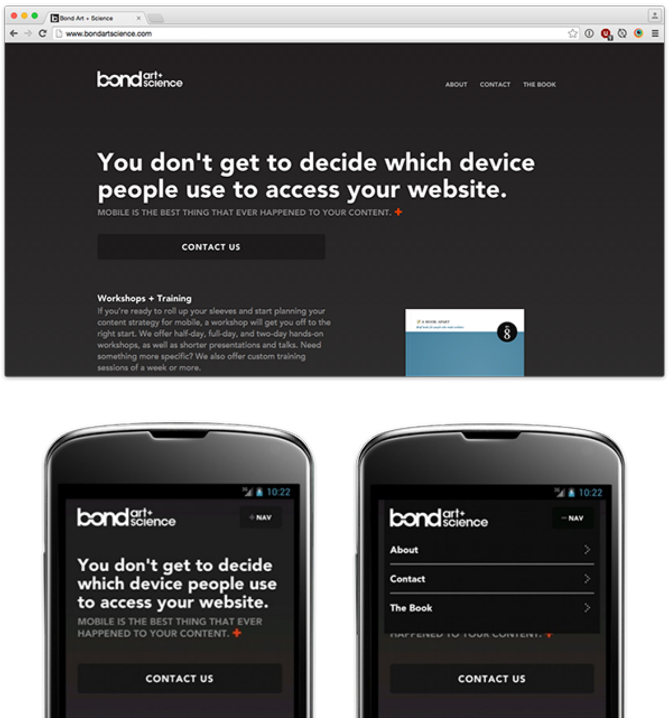
The Field Museum's responsive navigation, which occupies the height of the design. Below a certain width, the navigation moves to the top of the screen to avoid cropping.

On wider screens, the navigation is anchored to the left edge of the design, and spans the full height of the viewport. You may notice that they're using the flexible box model *(https://developer.mozilla.org/en-US/docs/Web/CSS/flex)*, or *flexbox*, an advanced CSS layout method we'll look at later in this chapter. But since flexbox allows elements to automatically fill the space available to them, as the menu gets taller or shorter, the navigation elements resize vertically—but below a certain width *or* height, the menu is placed at the top of the page.

By minding the navigation's vertical edges, the Field Museum was able to introduce alternate layouts to ensure the content inside their navigation menu was never concealed, obscured, or clipped. In other words, the breakpoints we introduce to our responsive designs aren't tied to the shape of a device's screen. Instead, our media queries defend the integrity of the content we're designing.

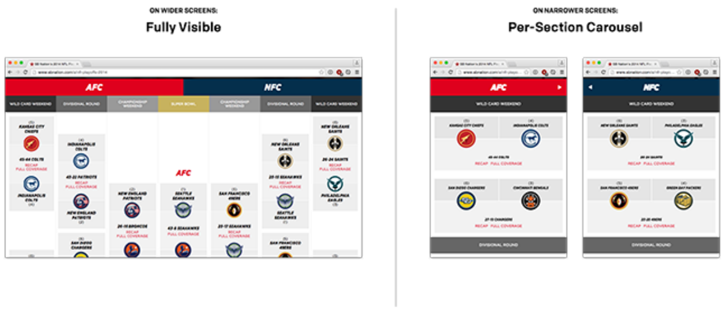
INTERACTION

The way we *interact* with an element may change along with the design. Responsive navigation systems are probably the most obvious example of this. As we saw in Chapter 2, menus are often displayed in full at wider breakpoints but concealed at smaller ones, perhaps hidden behind expandable icons or links when space is at a premium.


www.bondartscience.com
ABOUT
CONTACT
THE BOOK
You don't get to decide which device people use to access your website.
MOBILE IS THE BEST THING THAT EVER HAPPENED TO YOUR CONTENT.
CONTACT US
Workshops + Training
+ NAV
– NAV
About
Contact
The Book

Karen McGrane's company site has a traditional-looking navigation at wider breakpoints, but on smaller viewports the user toggles the visibility of the menu. Same links, but a new interaction model.

But navigation isn't the only kind of content that might require interaction changes. For example, take the responsive sports brackets designed by SB Nation *(http://bkaprt.com/rdpp/05-23/)*. While they appear as complex-looking charts at wider breakpoints, a simpler, more linear view of the brackets is shown on smaller screens.

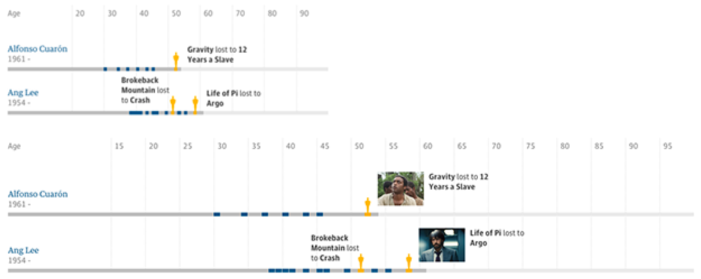


I don't know from sports, but I know I like SB Nation's responsive brackets: complex charts on wide screens, but a carousel of match-ups on smaller viewports. Same information, different interaction.

Along with the simplified layout, the brackets are presented as carousels in the smaller view, where real estate is more dear. Each of the four regions for the bracket are a single slide in the carousel, allowing the user to cycle through for details. The information in both visual states is the same, but the interaction model changes.

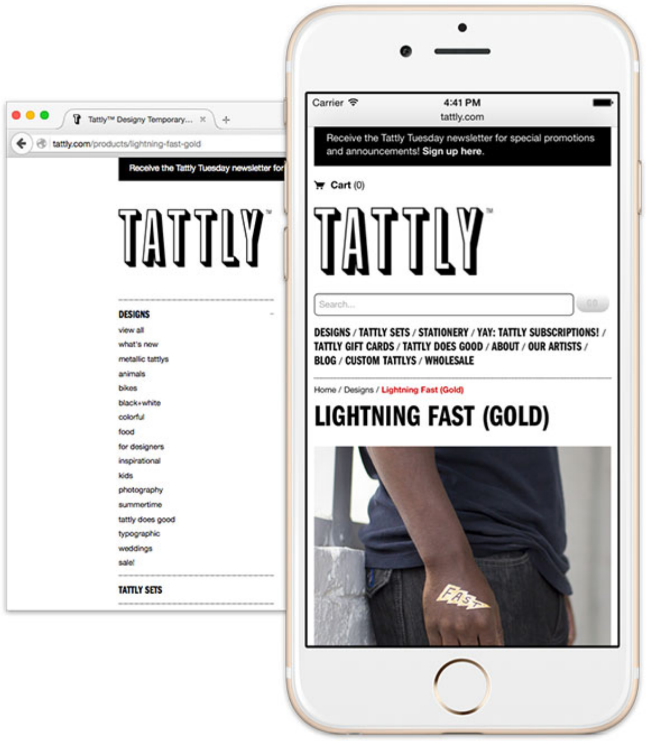
DENSITY

Finally, the amount of information you're showing in an element might need to vary over time—in other words, the *density* of information can change. The *Guardian*'s feature on the 2015 Oscars is full of examples of this, with responsively designed timelines displaying a significant amount of movie data. Above a certain width, thumbnail images are loaded in, slightly increasing the visual (and informational) density of the timeline.


Age
Alfonso Cuarón
1961 -
Gravity lost to 12 Years a Slave
Ang Lee
1954 -
Brokeback Mountain lost to Crash
Life of Pi lost to Argo

The Guardian's responsive cinematic timelines gradually increase in density, displaying an extra image above a certain width.

Density is, as you might have guessed, an area where you should tread very carefully. As we've discussed before, removing or hiding information because it doesn't fit can be problematic.



tattly.com/products/lightning-fast-gold
Receive the Tattly Tuesday newsletter for
TATTLY
DESIGNS
view all
what's new
metallic tattys
animals
bikes
black+white
colorful
food
for designers
inspirational
kids
photography
summertime
tattly does good
typographic
weddings
sale!
TATTLY SETS
Carrier
4:41 PM
tattly.com
Receive the Tattly Tuesday newsletter for special promotions and announcements! Sign up here.
Cart (0)
TATTLY
Search...
DESIGNS / TATTLY SETS / STATIONERY / YAY: TATTLY SUBSCRIPTIONS! / TATTLY GIFT CARDS / TATTLY DOES GOOD / ABOUT / OUR ARTISTS / BLOG / CUSTOM TATTLYS / WHOLESALE
Home / Designs / Lightning Fast (Gold)
LIGHTNING FAST (GOLD)

Tattly hides its submenu entirely, reducing its navigation to a list of primary sections on smaller screens.

Personally, I think the *Guardian*'s timelines work so well because the images shown at wider breakpoints are enhancements: they're not critical to understanding the information around them. Could they have designed alternate versions of the timelines to show images at all breakpoints? Possibly. But I think this is a wonderful example of density used to lighten the visual impact of a design, removing extraneous information without impeding access to the content.

About the Author

Ethan Marcotte

Ethan Marcotte is an independent web designer who cares deeply about beautiful design, elegant code, and the intersection of the two. Over the years, his clientele has included New York Magazine, the Sundance Film Festival, The Boston Globe, and People Magazine. Ethan can be found on Twitter or on his long-neglected

blog, and his most recent book is Responsive Web Design.

MORE FROM THIS AUTHOR

Fluid Images *(fluid-images)*, Responsive Web Design *(responsive-web-design)*, Fluid Grids *(fluidgrids)*, Where Our Standards Went Wrong *(whereourstandardswentwrong)*

ISSN 1534-0295 · Copyright © 1998–2016 A List Apart & Our Authors