

Responsive web design example created by Ethan Marcotte in his seminal article on RWD from 2010.

DESIGN APPROACHES FOR A RANGE OF DEVICES

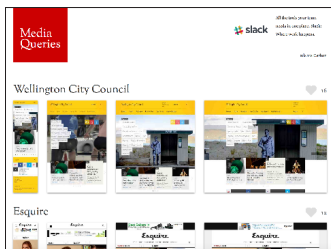
GRACEFUL DEGRADATION Designing for the fanciest browsers/viewing conditions, then making adjustments for less desirable viewing conditions (i.e., older, more limited browsers, etc.)

PROGRESSIVE ENHANCEMENT To design progressively is to ‘future-proof’ a design, building up from the perspective of the content, rather than from the perspective of browser performance.

AGILE PROCESS An approach to designing & building with rapid iteration and lots of show-and-tell with feedback loops. Assumes a systematic approach to content, rather than a page-based approach.

INGREDIENTS OF RESPONSIVE WEB DESIGN (RWD)

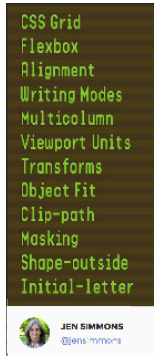
- 1) **FLEXIBLE LAYOUT** Size things in proportions whenever possible (ems and %), not pixels.
- 2) **FLEXIBLE CONTENT** Type, images and media (such as video) are set up and optimized for varied presentational conditions, especially sizing and proportioning.
- 3) **MEDIA QUERIES** Special CSS rules that enable you to make adjustments to any presentational aspect of your site design at “breakpoints,” that is, when the flexible layout changes too much due to severe adjustments in display size/proportion. The designer determines the breakpoints when media queries are used and adjusts certain CSS rules accordingly. Screen, print, projection and tv media can be targeted with media queries. Here is syntax for a simple, screen-specific, CSS media query example:



```
#wrapper {
width: 65%; }

@media screen and (max-width: 480px) {
#wrapper {
width: 100% }
}
```

Media Queries: a web database of well-designed, responsive web sites.



Jen Simmons' Exp. Layout Lab

INTRINSIC WEB DESIGN

LAYOUT APPROACHES Normal flow, CSS grid, flexbox, and multicolumn

INTRINSIC DESIGN IS CAPABLE OF:

- 1) CONTRACTING & EXPANDING** Elements of design, including white space, can be more organically designed into the visual form of pages layouts.
- 2) FLEXIBILITY** It is now possible to utilize a range of size designations for widths and heights of elements (alone or in combination) for maximum layout control. it's also now possible to automatically achieve some layout adjustments, minimizing the dependence on media queries, as in the case of the use of autofit and minmax():

```
#content-main {  
  display: grid;  
  grid-template-columns: repeat(auto-fit, minmax(320px, 1fr));  
}
```

- 3) NEW USES OF THE VIEWPORT** Vertical size and spacing control is now possible with grid and flex (especially vertical centering!).

HOW TO DESIGN AND BUILD RESPONSIVELY / INTRINSICALLY

- 1) THINK MOBILE FIRST.**
- 2) USE MODULAR THINKING.**
- 3) UNDERSTAND YOUR INFORMATION ARCHITECTURE.**
- 4) USE A FLUID/SYSTEMATIC STRUCTURE** Combine responsive and intrinsic techniques as needed. For responsive elements: use relative sizing, images/media and type, and media queries for big placement (or other) adjustments. For intrinsic elements: be lean with code to enable changes automatically.