http://hesketh.com/publications/progressive\_enhancement\_and\_the\_future\_of\_web\_design.html

Degrade with grace

## Cleaning House

that you should always work on a copy

The first thing we do is remove all presentational markup from our document.

Move scripts and stylesheet rulesets outside of the document.

## Mark It Up

## That Layered Look

The bugs can be used to selectively hide, or show, styles to whatever browser has that bug.

## Bugs are Your Friends

## Hack Once, Print Anywhere

For testing you can use basic browsers like a netscape navigator.

## Web Moderné

<http://uxmag.com/articles/combining-responsive-and-adaptive-strategies-to-solve-mobile-design-challenges>

responsive web design (RWD) adaptive web design (AWD, or mobile websites)

#### Responsive Web Design (RWD)

Pros

**The content stays the same.**

**It can be easily achieved automatically.**

**Cost**

Cons

**Generic not optimized experience.**

**No accounting for users’ behavioral differences with different mediums.**

**Loading time issues.**

#### Adaptive Web Design (AWD, or dedicated mobile websites)

Pros

**Speed.**

**Sensory design.**

Cons

**The content nightmare.**

**Too many versions.**

**Native app ‘transformation’.**

By combining RWD and AWD, however, you can often obtain the same user experience objectives far more cost effectively and efficiently than by using AWD or RWD alone.

<http://www.lukew.com/ff/entry.asp?1390>

Underlying each of these solutions is the same philosophy: give each device only what it needs using media queries and background images, JavaScript, or a server-side solution. This philosophy can dramatically cut down file size and increase performance.

For example, the Bagcheck desktop experience has all the information about a list of content, its comments, updates, and likes at a single URL. We bundle all of these sections (or modules) into a single file then load each section dynamically (as people request them) without a page refresh. While this creates a smooth transition on the desktop, it adds up to a lot of bytes on mobile.

So the mobile Web experience uses a different URL structure. The same URL loads the same initial content but each sub-section (comments, updates, likes) is a separate page with a unique URL as visualized in the image below.

<https://joshemerson.co.uk/blog/responsive-process/>

## A Responsive Workflow

## How Should We Respond?

## What Should We Respond To?

## Responsive Images

## Is Responsive Design Always Appropriate?

## The Responsive Summit

https://adaptivewebdesign.info

1. [Front Matter](https://adaptivewebdesign.info/1st-edition/read/index.html)
2. [Foreword](https://adaptivewebdesign.info/1st-edition/read/foreword.html)
3. [Chapter 1: Think of the User, Not the Browser](https://adaptivewebdesign.info/1st-edition/read/chapter-1.html)
4. [Chapter 2: Progressive Enhancement with Markup](https://adaptivewebdesign.info/1st-edition/read/chapter-2.html)
5. [Chapter 3: Progressive Enhancement with CSS](https://adaptivewebdesign.info/1st-edition/read/chapter-3.html)
6. [Chapter 4: Progressive Enhancement with JavaScript](https://adaptivewebdesign.info/1st-edition/read/chapter-4.html)
7. [Chapter 5: Progressive Enhancement for Accessibility](https://adaptivewebdesign.info/1st-edition/read/chapter-5.html)
8. [Chapter 6: Take It Away](https://adaptivewebdesign.info/1st-edition/read/chapter-6.html)
9. [Back Matter](https://adaptivewebdesign.info/1st-edition/read/back.html)

Fault tolerance - (отказоустойчивость) you can view any html page in any web browsers.

Graceful degradation was the philosophical equivalent of fault tolerance’s superficial.

### Tasty at any level

One analogy I like to use for progressive enhancement is the peanut M&M. At the center of a peanut M&M is, well, the peanut.

Technologies applied as layers—HTML, then HTML & CSS, then HTML, CSS & JavaScript—can create different experiences, each one equally valid (and tasty). And at the core of it all is the nut: great content.

Progressive enhancement isn’t about browsers. It’s about crafting experiences that serve your users by giving them access to content without technological restrictions. Progressive enhancement doesn’t require that you provide the same experience in different browsers, nor does it preclude you from using the latest and greatest technologies; it simply asks that you honor your content (and your users) by applying technologies in an intelligent way, layer-upon-layer, to craft an amazing experience. Browsers and technologies will come and go. Marrying progressive enhancement with your desire to be innovative and do incredible things in the browser is entirely possible, as long as you’re smart about your choices and don’t lose sight of your users.

### Progressive enhancement = excellent customer service

The baseline, first level of experience its most basic form: as text

The second level of experience comes from the semantics of the HTML language itself: emphasasing, bolding and etc.

The third level of experience is the audio-visual one, expressed through the use of CSS and the use of inline images, audio, and video.

The fourth level of experience is the interactive one. In the standards world, this level relies almost entirely on JavaScript

The final level is realized through the application of enhanced semantics and best practices contained within and used in conjunction with the Web Accessibility Initiative’s Accessible Rich Internet Applications (WAI-ARIA) spec.

