



# Advanced Topics

## Performance

- Why does performance matter?
  - Even 4G coverage is sparse for most of the world.
  - Assistive technology is impacted by performance.
  - Users have little patience for slow page loads.
  - Performance is a major component of UX.
  - Today's sites have gained weight. 3G was slow a few years ago, but many of today's sites seem useless on 3G.
  - A recent report from Google's DoubleClick revealed 53% of users abandon sites that take longer than 3 seconds to load.
  - Improved performance means increased engagement/conversion, which can translate to an increase in revenue. For example, Walmart saw a 2% increase in conversions for every 1 second improvement in page load. Every 100ms improvement resulted in a 1% increase in revenue.
- Optimization
  - Move scripts to the bottom of the page
  - `async` and `defer` your JavaScript files
  - Compress your images
  - Deliver the right assets for the device
  - Minify and concatenate your stylesheets and scripts
  - Prioritize Critical CSS, isolate CSS for above-the-fold content and inline it, to dramatically improve perceived performance. Using [Critical Path CSS Generator](#) can make this easy-peasy, or there's a Gulp task for that.
  - Then asynchronously load non-critical CSS using `rel="preload"` in your CSS link, with the `loadCSS polyfill`.
  - Move render-blocking CSS and JS to the bottom of the page.
  - Generate all your responsive image sizes and deliver in the `srcset` attribute
- Lazy Load content (intersection observers and otherwise)
- Web fonts impact on performance and UX

- By default @font-face has a flash of invisible text (FOIT) of up to 3 seconds while fonts load. In older -webkit browsers it can be even longer. If the font doesn't load, the users see nothing.
  - Better to embrace FOUT (flash of unstyled text) so users get content immediately and custom fonts swap in when available.
  - Use the new font-display property in your @font-face block, with values block, swap, fallback, and optional to instruct the browser how to behave while the web font downloads. Browser support is still lean, but it's **progressive enhancement**.
  - Alternatively, there's [Font Face Observer](#), a polyfill for the CSS Font Loading API
- Service Workers
    - Minimize network requests after the first visit
    - How does a Service Worker work?
    - Control which assets to cache (static, dynamic, etc), how to handle page requests, define the offline experience
    - More **progressive enhancement**!
    - Define it from scratch or use a tool like [Workbox](#).
    - However you do it, definitely do it, as the performance improvements are dramatic.
- Testing
    - [WebPagetest](#)
    - [Lighthouse](#)
    - [Speedcurve](#)
    - [PageSpeed Insights](#)

## HTTP2 and its impact in performance

## Microdata and structured markup

## Advanced examples using Gulp

- [Gulp Imagemin](#)

# Advanced examples using WebPack

## Pattern Libraries

- What they are?
- What are they good for?

## React, Preact, Angular, Vue — Oh, my! (libraries and frameworks)

- What they are?
- What are they good for?

## Web Components

- What they are?
- What are they good for?