DevExpress®

You *can* make fast web pages

Don Wibier
Technical Evangelist /DEVEXPRESS
Microsoft MVP

donw@devexpress.com
@donwibier

“Perceived” performance

- Design and Content
  - Typography, inter-relationships between whitespace and text, imagery
- Raw speed
  - Download speeds, rendering speed, avoiding “points of failure”

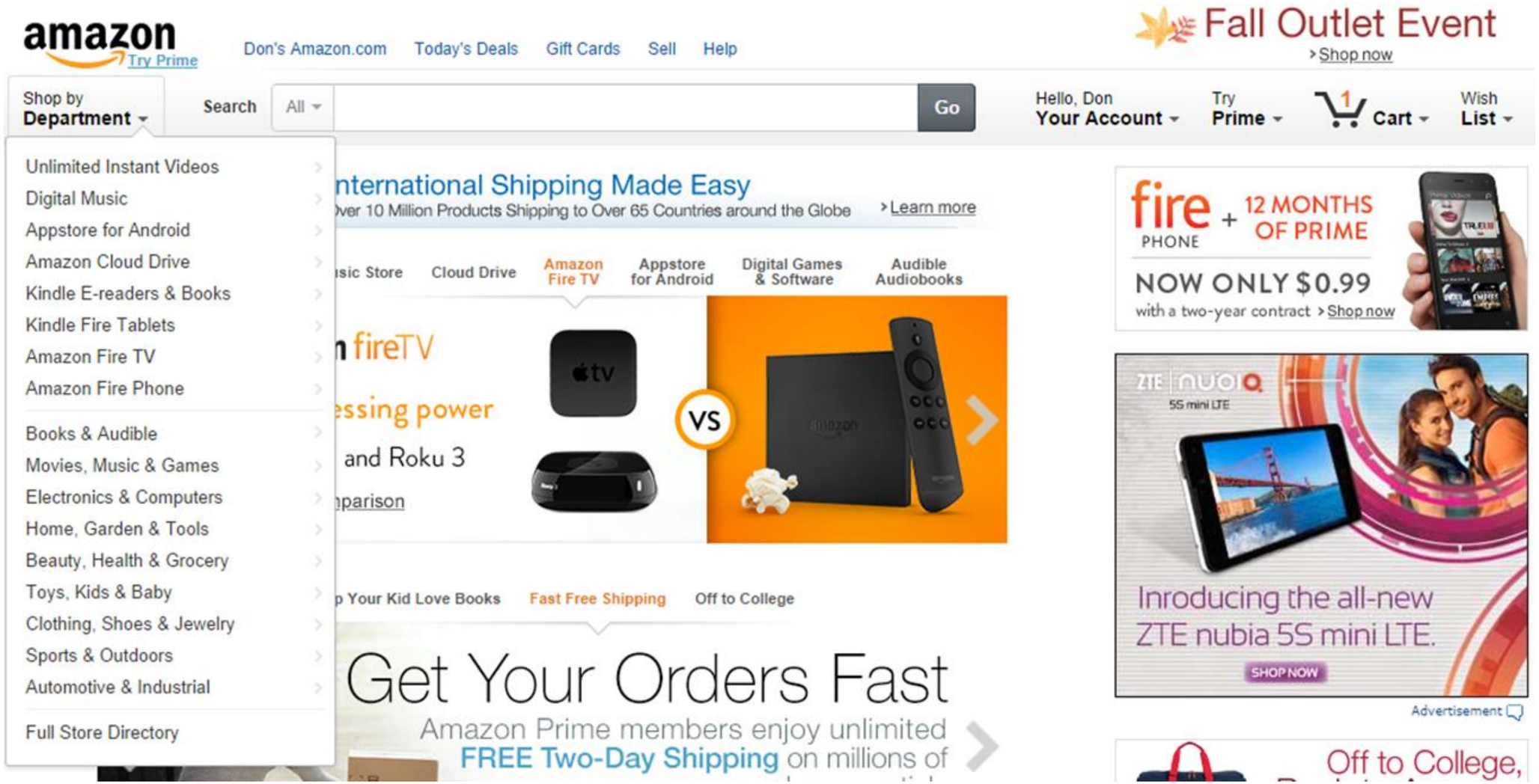
Breaking Bad

Web performance is like AstroJet Simulator

It's what the user *perceives* that's important

Breaking Bad

You **can** build fast web pages



DevExpress

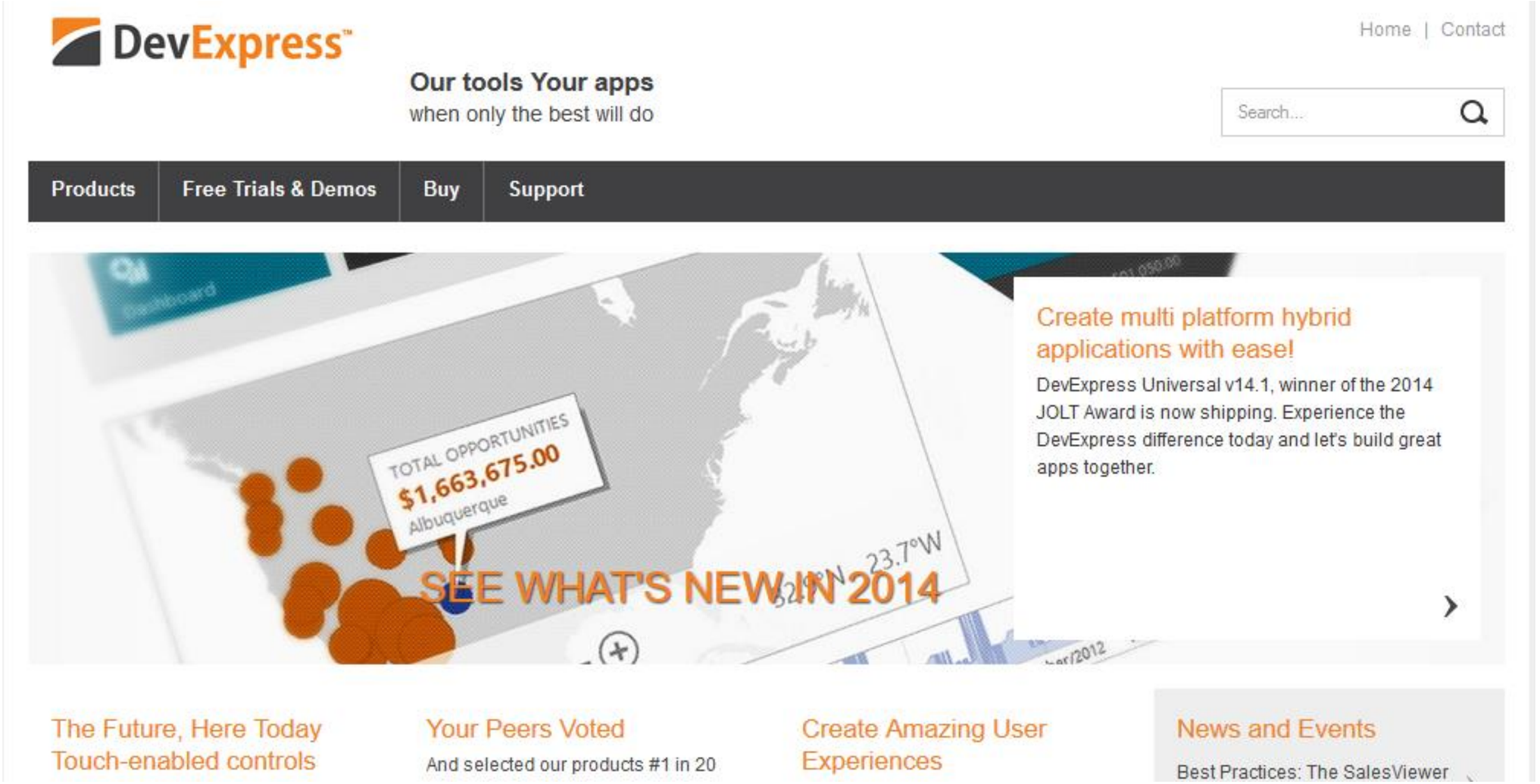
Nielsen's Response-Time limits

- 0.1 seconds
  gives the feeling of instantaneous response
- 1 second
  keeps the user's flow of thought seamless
- 10 seconds
  keeps the user's attention, but only just

source: http://dxpr.es/1iSTEvj

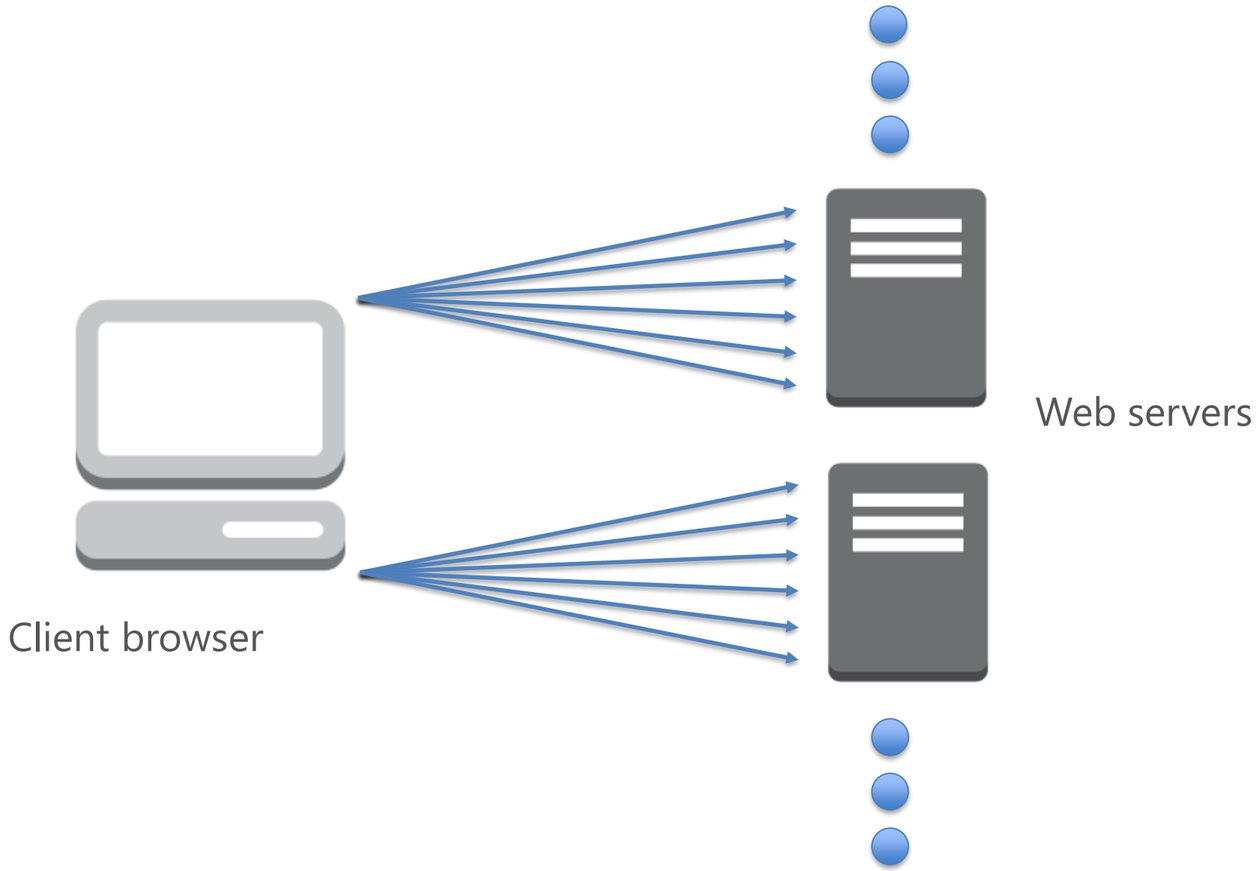
Breaking Bad

You **can** build fast web pages



DevExpress

Domain sharding



Sharding recommendation

Three hostnames max

- Main host for markup
- CDN for JavaScript files
- Cloud host for images

Reduce the number of files

Concatenate JavaScript and CSS files

- Caching issue
- Sharing issue
- Expiry Issue
- Strict Mode

Breaking Bad

JavaScript Strict Mode

- Insert “use strict” into code
  - Catches common errors
  - Throws exceptions for unsafe code
  - Disable dodgy features
- Apply to whole file of individual functions

```
"use strict";

/*
 * My JavaScript code
 */
function someFunction() {
    "use strict";
    var result = "";
    /*
     * rest of code
     */

}
```

Concatenating

- Test…
- Test…
- Test…

Minification

- Removes unneeded whitespace
- Renames internal identifiers to shorter name

Breaking Bad

You **can** build fast web pages

Minification

```
/*
 * My JavaScript code
 */
(function () {
    "use strict";
    window.myFunc = function (arr, separator, skipEmpty) {
            var result = '';
            if (arr && !skipEmpty) {
                result = arr.join(separator);
            }
            else {
                arr.forEach(function (value/*, index*/) {
                    var v = (value || '');
                    if (v.toString() !== '') {
                        result += (result !== '' ? separator : '') + v.toString();
                    }
                });
            }
            return result;
        };
}());
```

DevExpress

Breaking Bad

Minification

```
(function(){"use strict";window.myFunc=function(n,t,i){var r="";return n&&!i?r=n.join(t):n.forEach(function(n){var
i=n||"";i.toString()!==""&&(r+=(r!==""?t:"")+i.toString())}),r}})();
```

Minifier tools

- JSMin (granddaddy of minifiers)
- Yahoo YUI Compressor
- AjaxMin does JavaScript and CSS
- Grunt / Gulp Uglify tasks for JavaScript and CSS

Image optimization

- Add image width and height attributes to <img> tags

```
<img width="200px" height="150px" src="…" title="…" alt="…" />
```

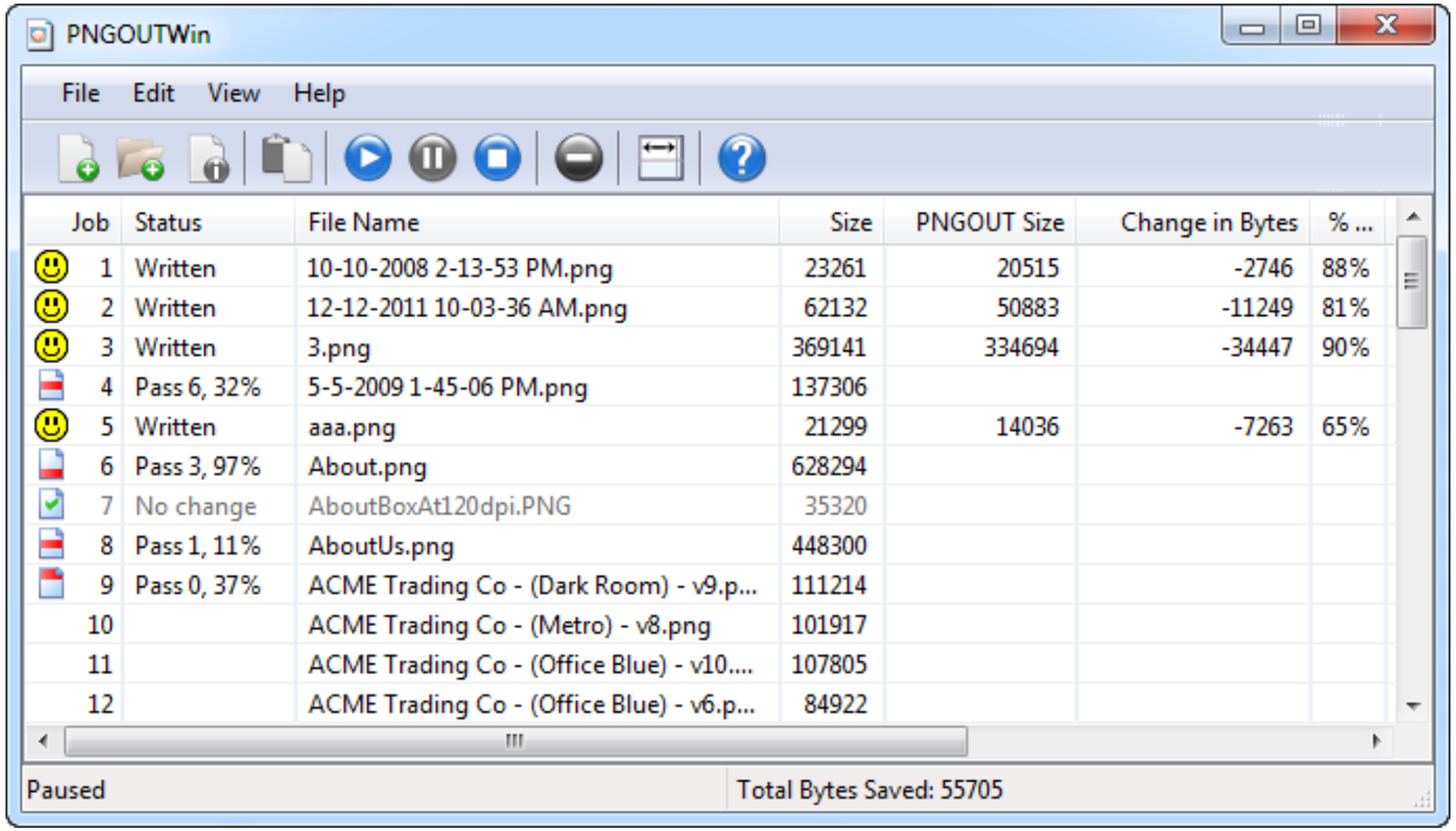
- Reduce image quality for JPGs
  Makes them fuzzier

- Recompress PNGs
  No quality loss, can take some time

- Concatenating images

Breaking Bad

PNG recompression with PNGOUT



DevExpress

Breaking Bad

Concatenating images ?

sprites.gif

```
#home {
    left: 0px;
    width: 46px;
    background: url('sprites.gif') 0 0;
}
#home a:hover {
    background: url('sprites.gif') 0 -45px;
}
#prev {
    left: 63px;
    width: 43px;
    background: url('sprites.gif') -47px 0;
}
#prev a:hover {
    background: url('sprites.gif') -47px -45px;
}
#next {
    left: 129px;
    width: 43px;
    background: url('sprites.gif') -91px 0;
}
#next a:hover {
    background: url('sprites.gif') -91px -45px;
}
```

```
<ul id="navlist">
  <li id="home"><a href="home.html"></a></li>
  <li id="prev"><a href="next.html"></a></li>
  <li id="next"><a href="prev.html"></a></li>
</ul>
```

Places scripts at the end

- Do not place <script> tags in the <head>
- Place them at the end of the <body>

Breaking Bad

Places scripts at the end

```
<!DOCTYPE html>

<html lang="en" xmlns="http://www.w3.org/1999/xhtml">
<head>
    <meta charset="utf-8" />
    <title></title>
    <link type="text/css" rel="stylesheet" href="css/mystyles.css" />
</head>
<body>
    <!-- your markup here -->
    <div class="container">
        <!-- ... -->
    </div>
    <script type="text/javascript" src="js/my-script.js"></script>
    <script type="text/javascript">
        doSomeCode();
        /* your javascript references and code
    </script>
</body>
</html>
```

Consider async and defer

- async
  - Load JavaScript asynchronously
  - Also means (possible) out-of-order execution
  - Good excuse for concatenation

```
<script async src="js/my-script.js"/>
```

- defer

  Load JavaScript after document has been processed

```
<script defer src="js/my-last-code.js" />
```

Consider a script loading library for async and defer

- WebPack
- HeadJS.com
- Requirejs.org

These tools take care of proper async and deferred script loading

Performance is subjective

- Improve **technical** speed

  But only attacking a small part of the problem

- Improve **perceived** performance

  Pay attention to the render time

Rule of thumb

- Assume 20% of time is spent downloading resources

- Rest of time (80%) is spent on rendering

Quick Recap

Try setting up a deployment script, and use browser dev tools

- Optimize caching
- Minimize round-trips
- Minimize download payloads
- Optimize rendering
- Test
- Test
- Test

Conclusion

Web performance is like AstroJet simulator

The important thing is how the user perceives the performance

Thank You!

donw@devexpress.com
@donwibier