Aaron Gustafson presents

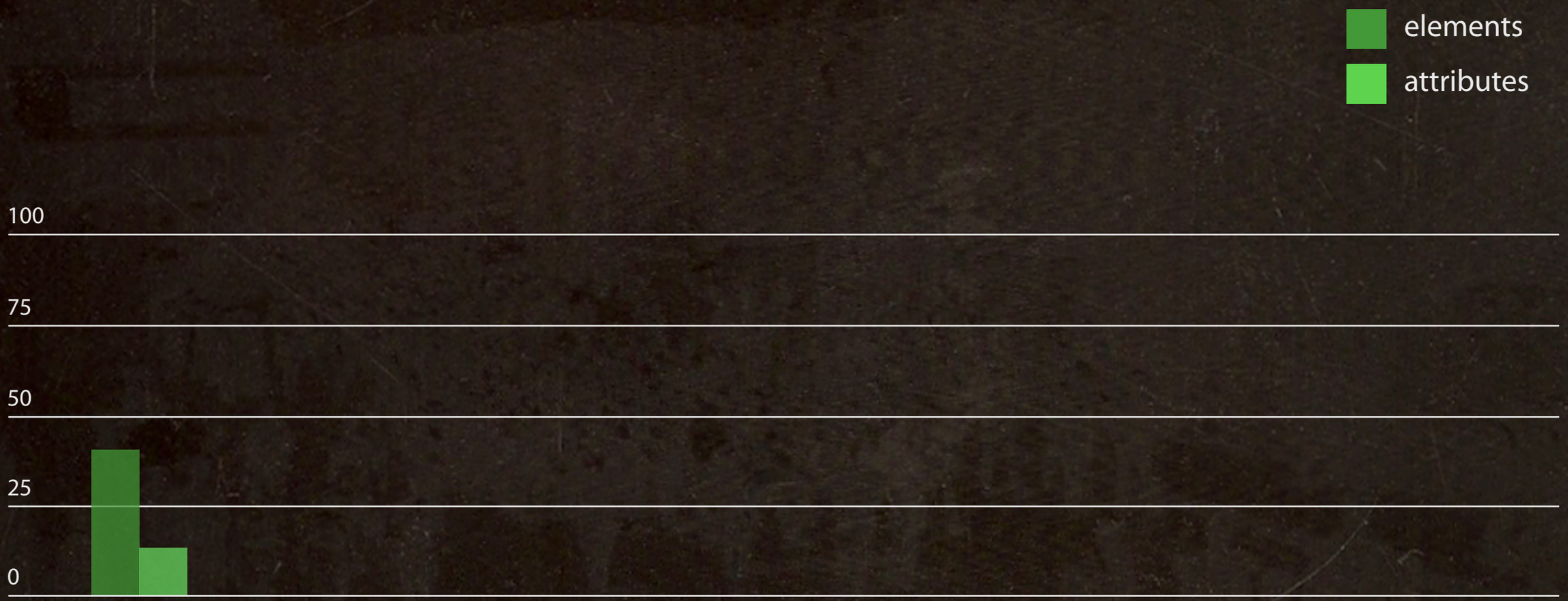
Solving CSS Problems with eCSStender

AN EVENT APART - CHICAGO 2009

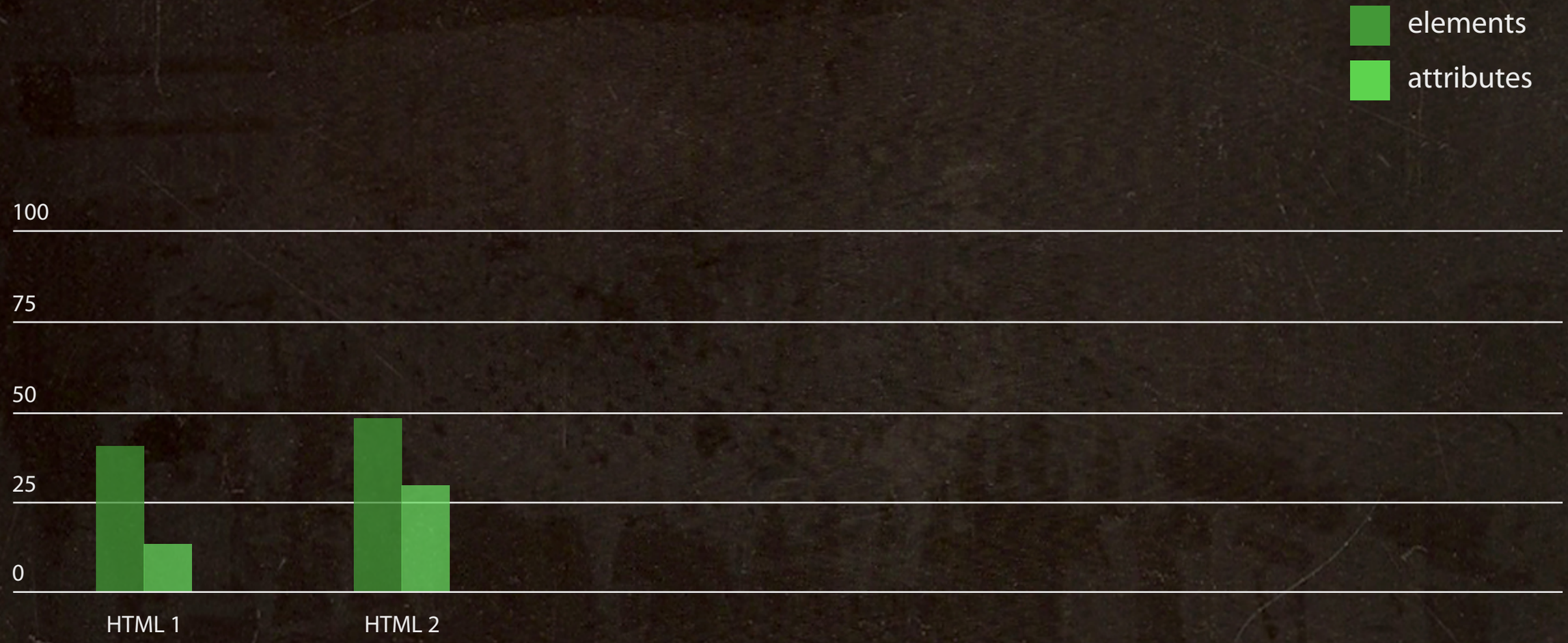
Where in the world is CSS3?

AN EVENT APART - CHICAGO 2009

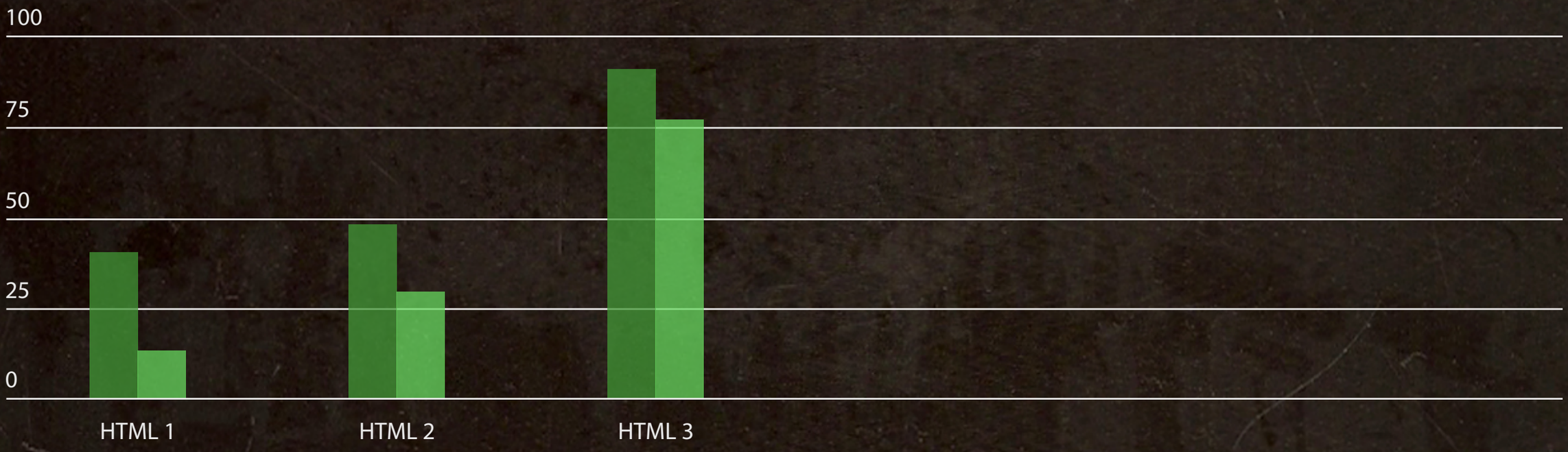
Progress



Progress

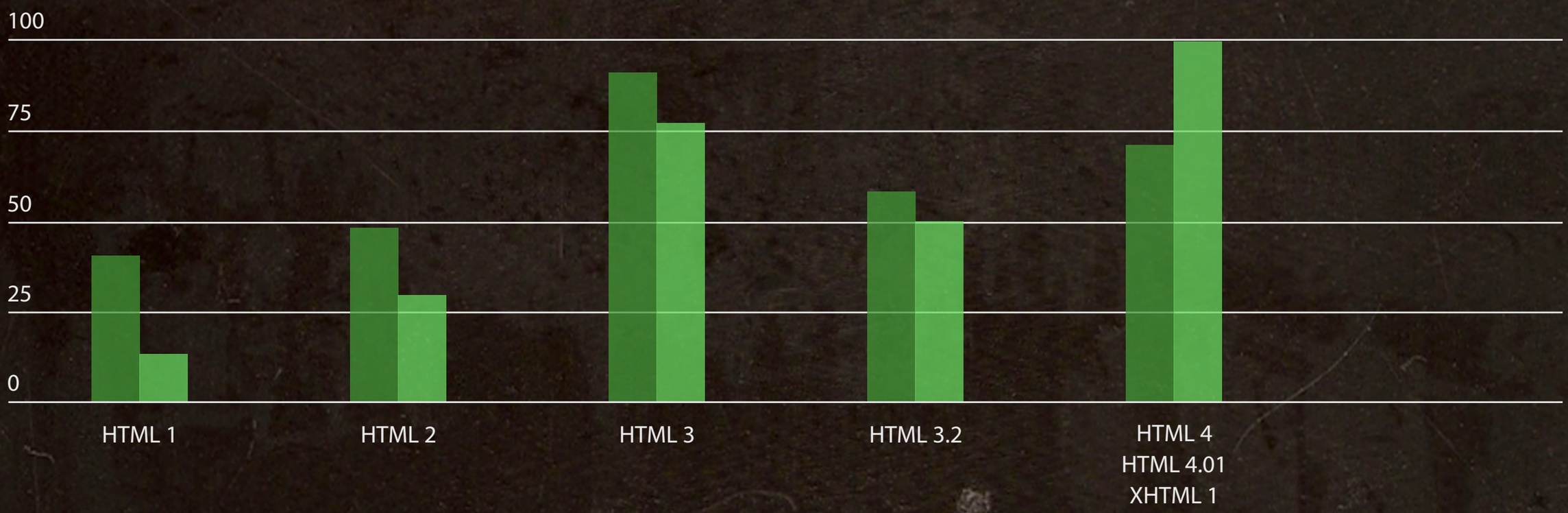


Progress



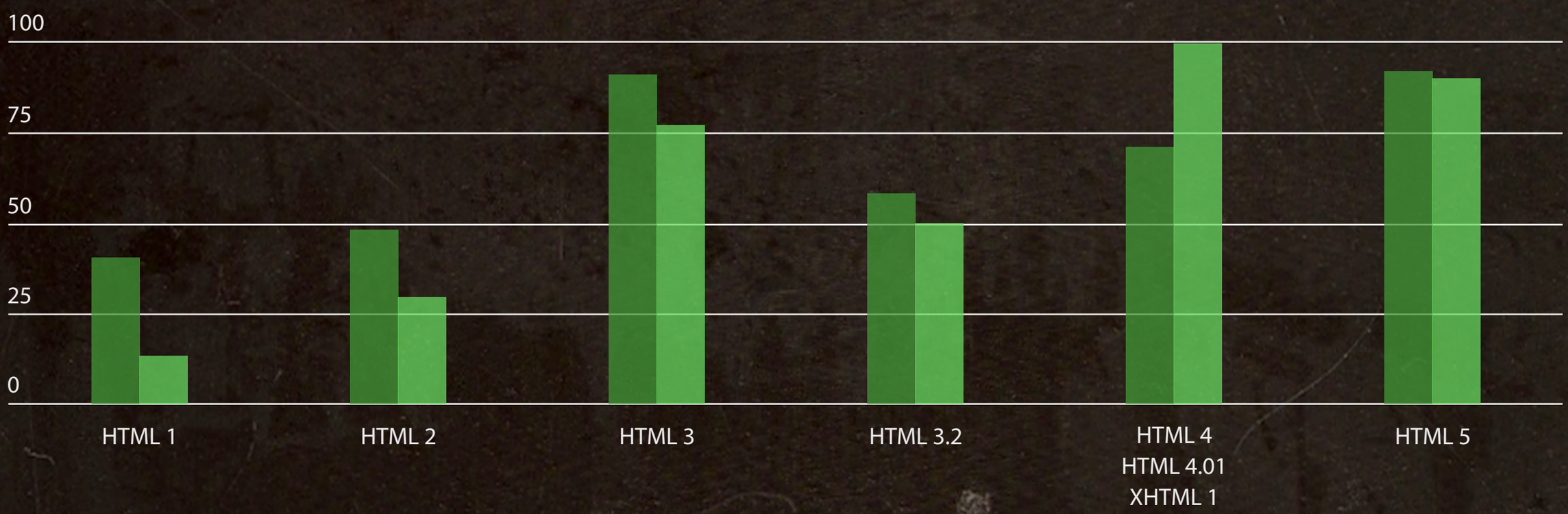
Progress

Progress



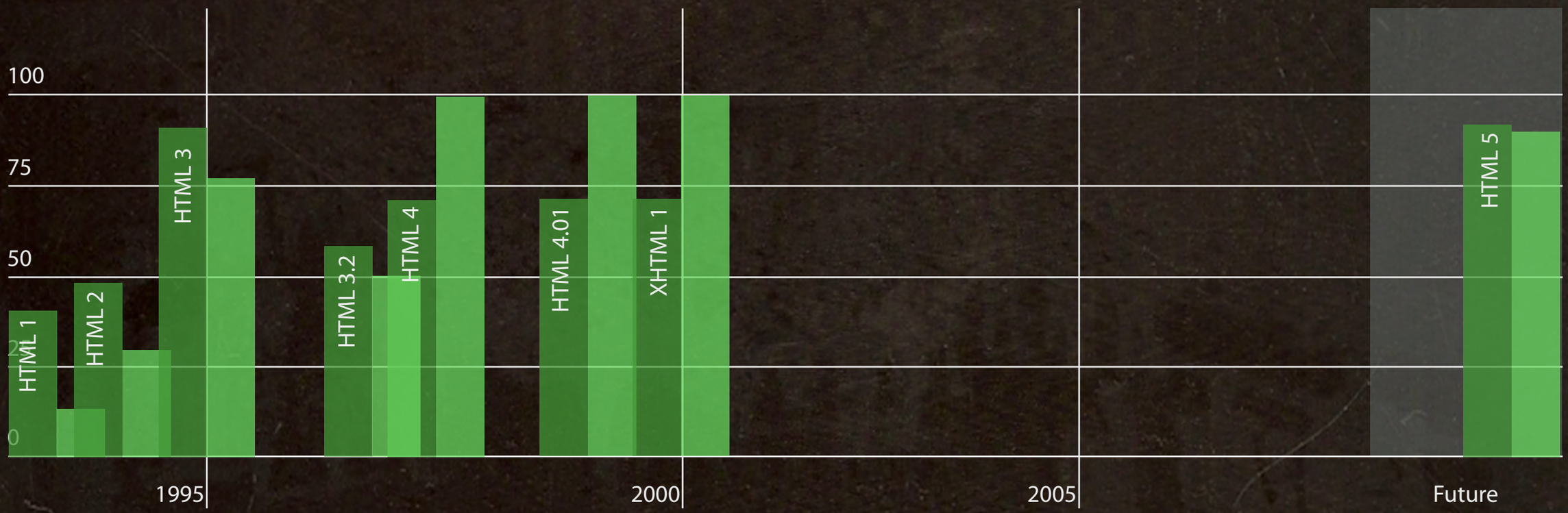
AN EVENT APART - CHICAGO 2009

Progress



AN EVENT APART - CHICAGO 2009

Progress?



AN EVENT APART - CHICAGO 2009

CSS 2.1 became a candidate recommendation last month

Some modules of CSS3 have been in development since 1999

Where does CSS come from?

AN EVENT APART - CHICAGO 2009

Officially?

AN EVENT APART - CHICAGO 2009

Officially?

Semi-officially

Microsoft®

AN EVENT APART - CHICAGO 2009

Officially?

Semi-officially

-moz-

-webkit-

Microsoft®

-ms-

-o-

Browsers “extend” CSS

opacity

border-radius

column-count

Browsers “extend” CSS

-moz-opacity

-moz-border-radius
-webkit-border-radius
-khtml-border-radius

-moz-column-count

Browsers “extend” CSS

```
#foo {
  border-radius: 3px;
}
```

Browsers “extend” CSS

```
#foo {
  -moz-border-radius: 3px;
  -webkit-border-radius: 3px;
  -o-border-radius: 3px;
  -khtml-border-radius: 3px;
}
```

Browsers “extend” CSS

```
#foo {
  border-radius: 10px 5px;
}
```

Browsers “extend” CSS

```
#foo {
  -moz-border-radius: 10px 5px;
  -webkit-border-top-left-radius: 10px;
  -webkit-border-top-right-radius: 5px;
  -webkit-border-bottom-right-radius: 10px;
  -webkit-border-bottom-left-radius: 10px;
  -o-border-radius: 10px 5px;
  -khtml-border-top-left-radius: 10px;
  -khtml-border-top-right-radius: 5px;
  -khtml-border-bottom-right-radius: 10px;
  -khtml-border-bottom-left-radius: 10px;
}
```

Browsers “extend” CSS

```
#foo {
  padding: 10px;
  width: 200px;
  w\idth: 180px;
  height: 200px;
  heigh\t: 180px;
}

/* or */

#foo {
  padding: 10px;
  width: 200px;
  height: 200px;
}
* html #foo {
  width: 180px;
  height: 180px;
}
```

Can we play too?

AN EVENT APART - CHICAGO 2009

First Experiment: 2005

```
#content:before {
  -gfss-source: url(spiral.swf);
  -gfss-background-color: #000;
  -gfss-flash-version: 6;
  -gfss-quality: high;
  -gfss-flash-vars: "foo=bar&bar=foo";
  -gfss-width: 50;
  -gfss-height: 50;
  position: absolute;
  top: 0;
  left: 0;
}
```

First Experiment: 2005

3 years later…

AN EVENT APART - CHICAGO 2009

{+} ecsstender

AN EVENT APART - CHICAGO 2009

Fast facts

- It's pronounced "extender"
- A small JS library (<16K minified) for interfacing with CSS
- Is library agnostic (plays well with anyone)
- MIT License
- Enables both
  - the creation of CSS extensions to push the boundaries of the language (experimentally or practically), and
  - the patching of old/outdated browsers to the latest standards
- Allows designers to include extensions in their site and have their CSS **Just Work**.

Fast facts

- It's media aware (including `@media`)
- It can handle both linked and imported CSS (nesting too)
- It understands concepts like `@font-face` and `@page`
- It runs on DOM ready (undetectably in many cases)
- Its CSS parser functions much like a browser's, paying attention to the two key concepts:
  - Specificity
  - Cascade
- It supports local storage (`window.localStorage` and IE's `UserData`) for client-side caching

How does it work?

```
eCSStender.register(
  { selector: 'h1' },
  '*',
  function( selector, properties, media, specificity ){
    // do something
});
```

How does it work?

```
eCSStender.register(
  { selector: 'h1' },
  '*',
  function( selector, properties, media, specificity ){
    // do something
});
```

extension registration

How does it work?

```
eCSStender.register(
  { selector: 'h1' },
  '*',
  function( selector, properties, media, specificity ){
    // do something
});
```

define the extension

How does it work?

```
eCSStender.register(
  { selector: 'h1' },
  '*',
  function( selector, properties, media, specificity ){
    // do something
});
```

property request

How does it work?

```
eCSStender.register(
  { selector: 'h1' },
  '*',
  function( selector, properties, media, specificity ){
    // do something
});
```

callback function

In prose

- Search for a selector containing “h1”
- Collect every property set on it
- Send each selector, its properties, media type and specificity to my function so I can do something with it.

```
eCSStender.register(
  { selector: 'h1' },
  '*',
  function( selector, properties, media, specificity ){
    // do something
  });
```

Do you want to see something cool?

AN EVENT APART - CHICAGO 2009

```
#sidebar {
  width: 30%;
  float: right;
  position: relative;
  transform-origin: 0 0;
  transform: rotate(-15);
}
```

AN EVENT APART - CHICAGO 2009

eCSStender Example: transform: rotate()

Easy!%20Designs,%20LLC/Projects/Code/eCSStender.js/examples/transform.html

Google

Clients ▾ Easy! Stuff ▾ Personal ▾ Localhost ▾ Favelets ▾ Resources ▾ CSS3 Selectors Module CSS Stuff Bikes Clever PNG ...ng Magazine

eCSStender Example: transform: r...

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi.

Header	Data	Data	Data
Header	Data	Data	Data
Header	Data	Data	Data
Header	Data	Data	Data
Header	Data	Data	Data

Lorem ipsum dolor sit amet, HTML consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi.

0 errors / 0 warnings 0 errors / 0 warnings

eCSStender Example: transform: rotate()

file:///Users/aaron/Easy! Designs, LLC/Projects/Code/eCSStender.js/

Clients | Easy! Stuff | Personal | Localhost | Favelets | Resources | CSS3 Selectors Mo... | CSS Stuff | Bikes | Clever PNG Optimiza... | Green Power Switch

Addresses | Contacts | Events | Locations | Tagspaces | Bookmarks | Resources | Options

eCSStender Example: transfor...

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi.

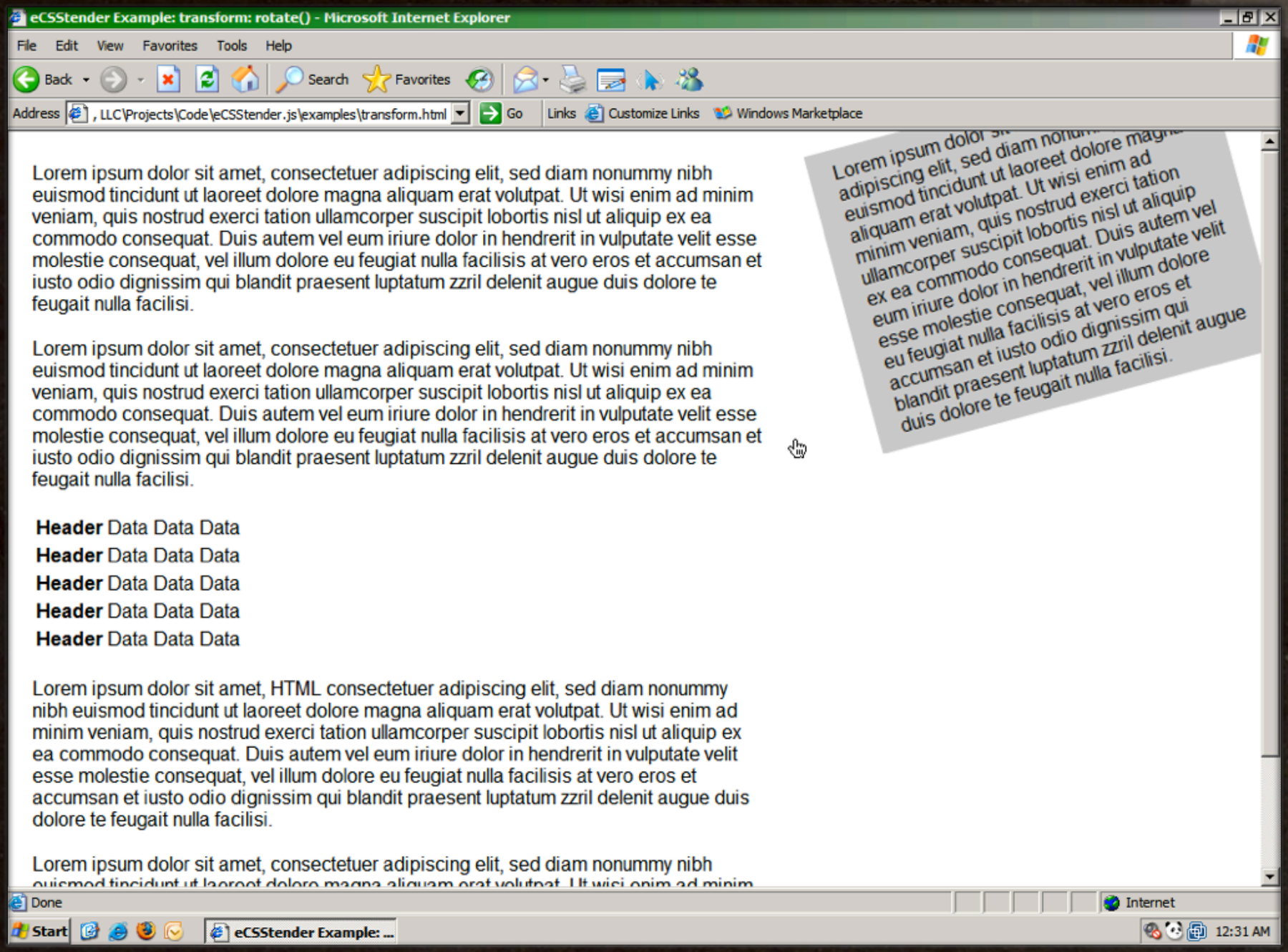
Header	Data	Data	Data
Header	Data	Data	Data
Header	Data	Data	Data
Header	Data	Data	Data
Header	Data	Data	Data

Lorem ipsum dolor sit amet, HTML consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut

Lorem ipsum dolor sit amet, adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugai nulla facilisi.

Done

YSlow


eCSStender Example: transform: rotate() - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address , LLC\Projects\Code\eCSStender.js\examples\transform.html
Go
Links Customize Links Windows Marketplace
Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi.
Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi.
Header Data Data Data
Header Data Data Data
Header Data Data Data
Header Data Data Data
Header Data Data Data
Lorem ipsum dolor sit amet, HTML consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi.
Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh
Done
Internet
Start
eCSStender Example: ...
12:31 AM

Finding what you want

AN EVENT APART - CHICAGO 2009

The definition

- Requires a lookup method
  - `selector`
    - simple selector string: `{ selector: 'h1' }`
    - compound selector string: `{ selector: 'h1, h2' }`
    - regular expression: `{ selector: /\s*h1\s*/ }`
    - array of any of the above: `{ selector: [ 'h1', /\s*h1\s*/ ] }`
    - a selector function:

```
{
  selector: function(){
    return ( this.match(/\s*h1\s*/) !== null );
  }
}
```

The definition

- Requires a lookup method
  - `property`
    - simple string: `{ property: 'border-radius' }`
    - array of strings: `{ property: [ 'border-radius' ] }`
  - `fragment` `{ fragment: 'radius' }`
  - `prefix` `{ prefix: 'moz' }`

The definition

- Can be
  - media-restricted:

```
{
  selector: 'h1',
  media: 'screen, projection'
}
```

  - filtered by property name or value:

```
{
  property: 'transform',
  filter: {
    value: /rotate\(-?\d+?\)/
  }
}
```

The definition

- Can be
  - fingerprinted:

```
{
  fragment:    'radius',
  fingerprint: 'net.easy-designs.border-radius'
}
```

Getting what you want

AN EVENT APART - CHICAGO 2009

Properties, please.

- eCSStender tries to be smart about properties you want
  - property-based lookups (property, fragment, prefix) **always** return the matching properties, allowing you to supply `false` as the second argument of `register()`

```
eCSStender.register(
  {
    fragment:    'radius',
    fingerprint: 'net.easy-designs.border-radius'
  },
  false,
  function(){});
```

Properties, please.

- You can still specify what you want though
  - Get every property on the matching element using "*"

```
eCSStender.register(
 { selector: 'h1' },
 '*',
 function(){});
```

  - or you can specify the properties you want using a string:

```
eCSStender.register(
 { selector: 'h1' },
 'color',
 function(){});
```

or

```
eCSStender.register(
 { selector: 'h1' },
 ['color','visibility'],
 function(){});
```

Doing something with what you find

AN EVENT APART - CHICAGO 2009

The callback

- When matches are found, your callback function is run
- eCSStender supplies your callback function with the following four arguments (in order)
  - **selector** - the selector that was matched
  - **properties** - a hash of the desired properties from a match
  - **medium** - the medium in which the match was found
  - **specificity** - a sum-total of the selector's specificity

The callback

```
eCSStender.register(
 { selector: 'h1' },
 '*',
 function( selector, properties, medium, specificity ){

   alert( 'my extension found ' + selector );

   if ( typeof properties['color'] != 'undefined' )
   {
     alert ( 'this h1 has a color!' );
   }

   alert( 'the medium is ' + medium );

   alert( 'it has a specificity of ' + specificity );

 });
```

Testing Testing 1-2-3

AN EVENT APART - CHICAGO 2009

It's wishful thinking...

- but what happens when browsers start doing things correctly?
- eCSStender helps you prepare by supporting robust testing for selector and property support: `eCSStender.isSupported()`
  - test for property support*:

```
eCSStender.isSupported( 'property',
                        'border-top-left-radius: 3px' );
```

It's wishful thinking...

- but what happens when browsers start doing things correctly?
- eCSStender helps you prepare by supporting robust testing for selector and property support: `eCSStender.isSupported()`
  - test for property support*:

```
eCSStender.isSupported( 'property',
                        'border-top-left-radius: 3px' );
```

* pay attention to how browsers store values!

It's wishful thinking...

- but what happens when browsers start doing things correctly?
- eCSStender helps you prepare by supporting robust testing for selector and property support: `eCSStender.isSupported()`
  - test for selector support:

```
var
div  = document.createElement('div'),
p    = document.createElement('p'),
p2   = p.cloneNode(true);
div.appendChild( p ); div.appendChild( p2 );
eCSStender.isSupported( 'selector', 'div p ~ p',
                        div, p2 );
```

html context

element that this should select

Where to test

- Testing can be useful in many places
  - The `test` property of your extension definition (`true` == run):

```
eCSStender.register(
  { selector: /:empty/,
    test:     function(){
      var
      div  = document.createElement('div'),
      p    = document.createElement('p');
      div.appendChild( p );
      return ( ! eCSStender.isSupported(
        'selector', 'div p:empty', div, p ) );
    }
},
...
```

Where to test

- Testing can be useful in many places
  - Within your callback function:

```
eCSStender.register( ... function( ... ){ ...
    if ( ( eCSStender.isSupported( 'selector',
            'p:nth-child(odd)', div, p ) &&
          ! eCSStender.isSupported( 'selector',
            'p:nth-child(2n+1)', div, p ) &&
        selector.match(
          /:nth-child\(\s*(?:even|odd)\s*\)/ ) != null )
       || eCSStender.isSupported( 'selector',
            'p:nth-child(2n+1)', div, p ) ){
      // support is incomplete, but fixable
    } else {
      // we need to compensate for no support
    }
});
```

Helpers

AN EVENT APART - CHICAGO 2009

Working with styles

- Basic premise in eCSStender: applying inline style sucks and should be avoided because it
  - has the greatest specificity
  - runs roughshod over the cascade

Working with styles

- eCSStender encourages other methods of style application:
  - `eCSStender.embedCSS()`
    - takes 2 arguments: the styles to apply and (optionally) the medium to which to apply them

```
var styles = 'h1 { color: red; } h2 { color: pink; }';
eCSStender.embedCSS( styles );
```

or

```
var styles = 'h1 { color: red; } h2 { color: pink; }';
eCSStender.embedCSS( styles, 'screen' );
```

Working with styles

- eCSStender encourages other methods of style application:
  - `eCSStender.newStyleElement()`
    - allows for the creation of your own embedded stylesheet
    - takes two optional string arguments: the medium and `id` to apply
    - returns a reference to the `style` element
  - `eCSStender.addRules()`
    - takes two arguments: the `style` element and the styles to add

```
var myCSS = eCSStender.newStyleElement(),
styles = 'h1 { color: red; } h2 { color: pink; }';
eCSStender.addRules( myCSS, styles );
```

Working with styles

- Sometimes, however, you can't avoid applying inline style, so eCSStender tries to make it better
- `eCSStender.applyWeightedStyle()` takes 3 arguments:
  - a reference to the element
  - an JSON object containing the properties to apply
  - the specificity with which to apply them
  - keeps track of the specificity with which each property is applied in order to facilitate smarter inline style application

Working with styles

- Sometimes, however, you can't avoid applying inline style, so eCSStender tries to make it better

```
eCSStender.applyWeightedStyle( el,
  { 'visibility': 'hidden' }, 10 );  // el is hidden

eCSStender.applyWeightedStyle( el,
  { 'visibility': 'visible' }, 1 );  // el is still hidden

eCSStender.applyWeightedStyle( el,
  { 'visibility': 'visible' }, 10 ); // el is visible

eCSStender.applyWeightedStyle( el,
  { 'visibility': 'hidden' }, 100 ); // el is hidden again
```

Fonts and pages

- eCSStender exposes various "@" rules, including
  - `@font-face` as an object in the `eCSStender.fonts` array

CSS:

```
@font-face {
   font-family: 'Comfortaa';
   font-weight: normal;
   src: url('/fonts/Comfortaa_Regular');
}
```

`eCSStender.fonts`:

```
[ { 'font-family': 'Comfortaa',
    'font-weight': 'normal',
      src:          "url('/fonts/Comfortaa_Regular')" } ]
```

Fonts and pages

- eCSStender exposes various "@" rules, including
  - `@page` as a property of the `eCSStender.pages` object

CSS:

```
@page {
  margin: 3cm;
}
@page :right {
  margin-left: 5cm;
}
```

`eCSStender.pages`:

```
{
  all: { margin: '3cm' },
  right: { 'margin-left': '5cm' }
}
```

Performance

- eCSStender tries to be smart about which stylesheets it pays attention to and which it doesn't
  - foreign stylesheets are always ignored
  - you can tell eCSStender to ignore specific on-site stylesheets

```
eCSStender.ignore('foo.css');
eCSStender.ignore(['foo.css', 'bar.css']);
```

- eCSStender will attempt to cache extensions and the rules they apply to automatically, but you can disable that

```
eCSStender.disableCache();
```

Advanced eCSStender

- Use `eCSStender.lookup()` to gain introspection into the stylesheets from your callback function
  - lookup by selector, property, fragment or prefix
  - restrict results by medium and/or specificity

```
matches = eCSStender.lookup(
  {
    selector:    'h1',
    specificity: 0     // max specificity
  },
  '*'
); // matches.length === 0
```

Advanced eCSStender

- Use `eCSStender.lookup()` to gain introspection into the stylesheets from your callback function
  - lookup by selector, property, fragment or prefix
  - restrict results by medium and/or specificity

```
matches = eCSStender.lookup(
  {
    selector:    'h1',
    specificity: { min: 0, max: 2 }
  },
  '*'
);
```

Advanced eCSStender

- Use `eCSStender.lookup()` to gain introspection into the stylesheets from your callback function
  - lookup by selector, property, fragment or prefix
  - restrict results by medium and/or specificity
  - returns an array of style objects:

```
[ { medium:      'all'
    selector:    'h1',
    specificity: 1
    properties:  { color: 'red' }
  }, ... ]
```

Advanced eCSStender

- Create new eCSStender methods

```
eCSStender.addMethod('myMethod',function(){
  alert('hello');
});
eCSStender.methods.myMethod();
eCSStender.methods['myMethod']();
```

- Supply custom functions to be run when eCSStender finishes

```
eCSStender.onComplete(function({
  alert('eCSStender is finished processing!');
});
```

Let's make an extension

AN EVENT APART - CHICAGO 2009

Solving CSS Problems with eCSStender

by Aaron Gustafson

Slides available at
http://slideshare.net/AaronGustafson

eCSStender
http://eCSStender.org

flickr Photo Credits

“Photo plate 5” by SophieG*
“World Map 1689 — No. 1” by Caveman 92223 — Great to be Home
“Calendar Card - January” by Joe Lanman
“Wheeled School Bus - Right Front” by Bill Ward's Brickpile
“Röck!! on the right” by Adactio
http://flickr.com/photos/aarongustafson/galleries/72157622372250877/

AN EVENT APART - CHICAGO 2009