State of the Web
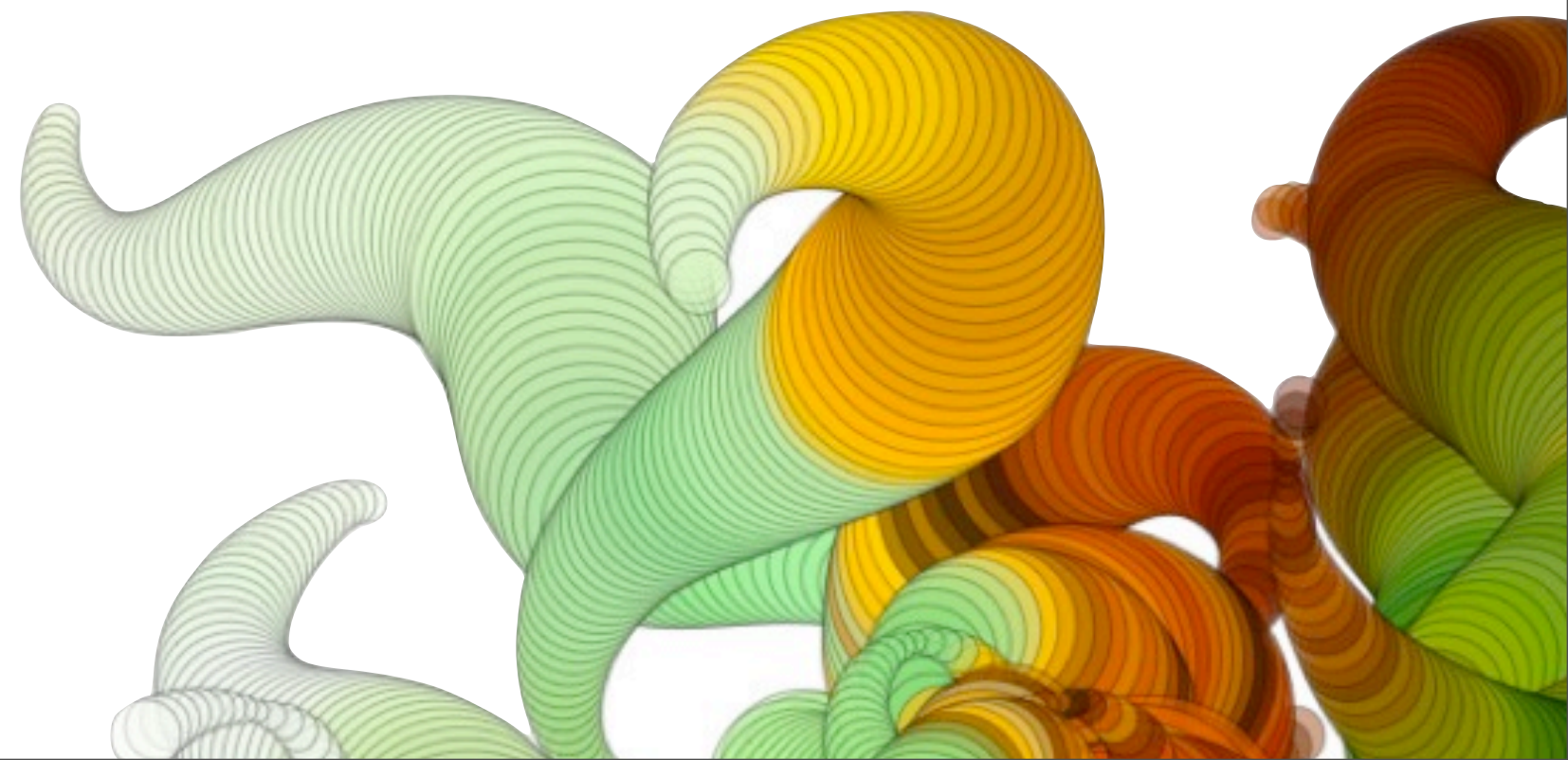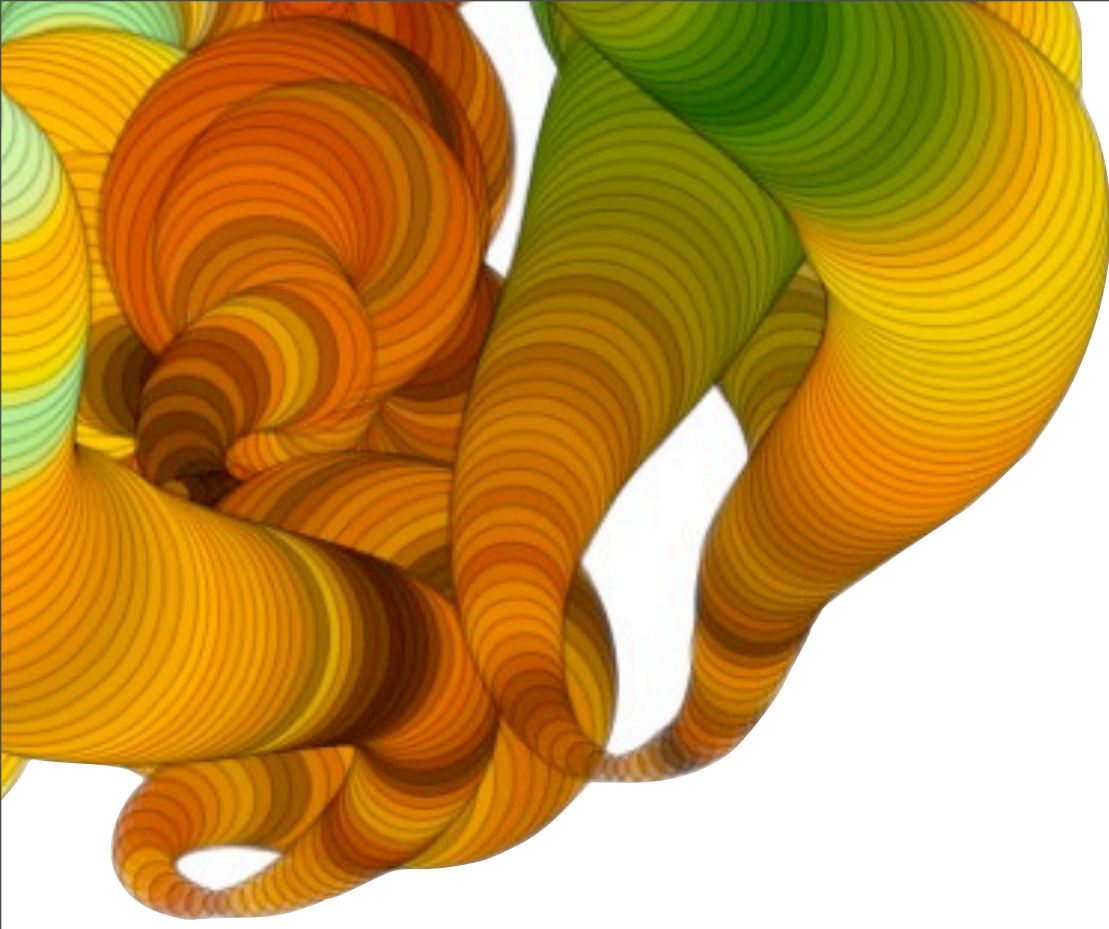
Mike Chambers, Adobe

HTML5 Camp Tokyo
August 3, 2011

What is HTML5? What do we mean by HTML5?

–Very specific specification, that covers a lot of new features
–Also used as a blanket term for new features being added to browsers.
    –CSS3
    –Features that have their own specifications (i.e. data storage)
    –Even sometime older features such as media queries and SVG

What is driving the growth of HTML5?

Browser competition has driven HTML5 by prompting browser vendors to innovate to stay competitive and relevant.
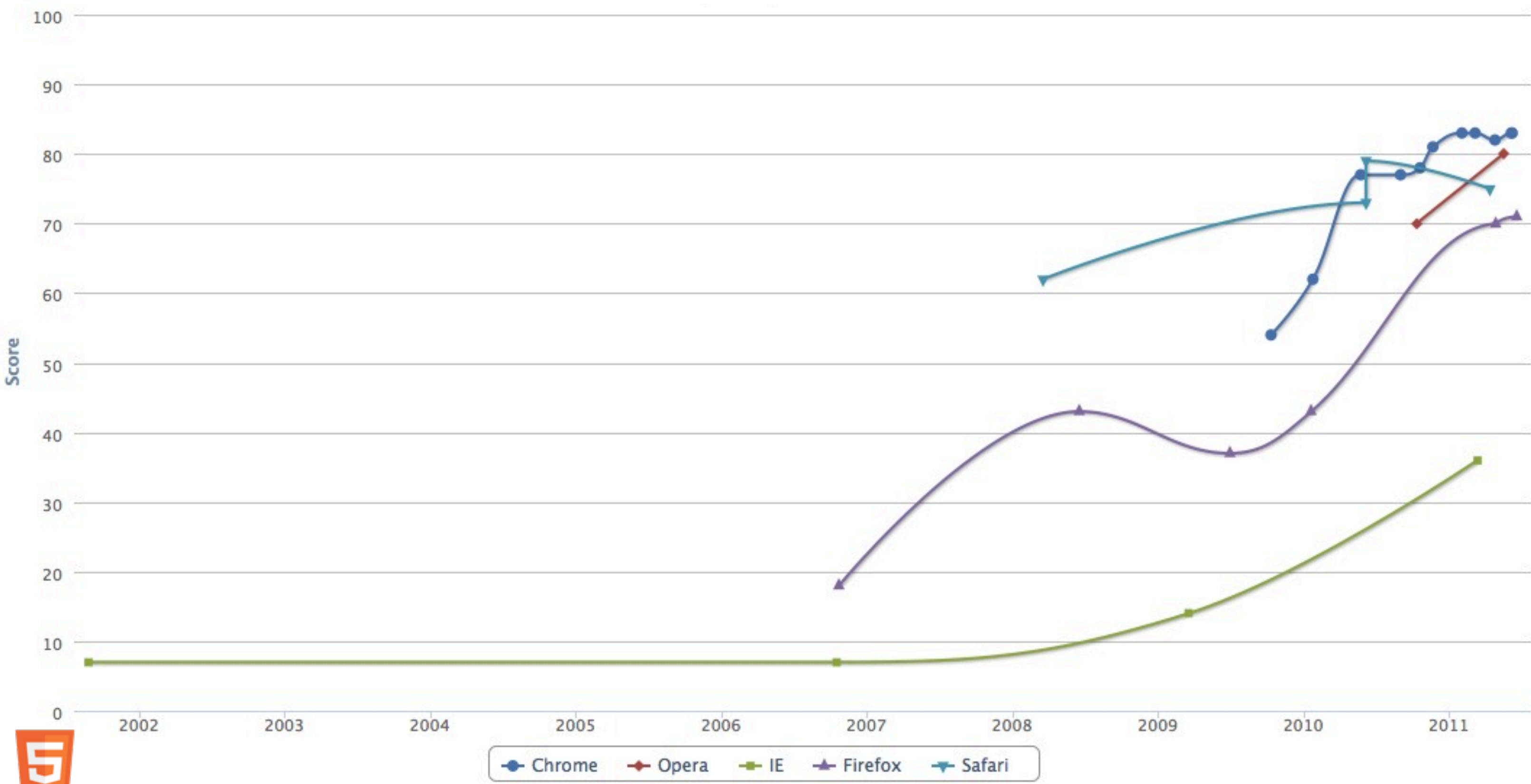
Usage share of web browsers
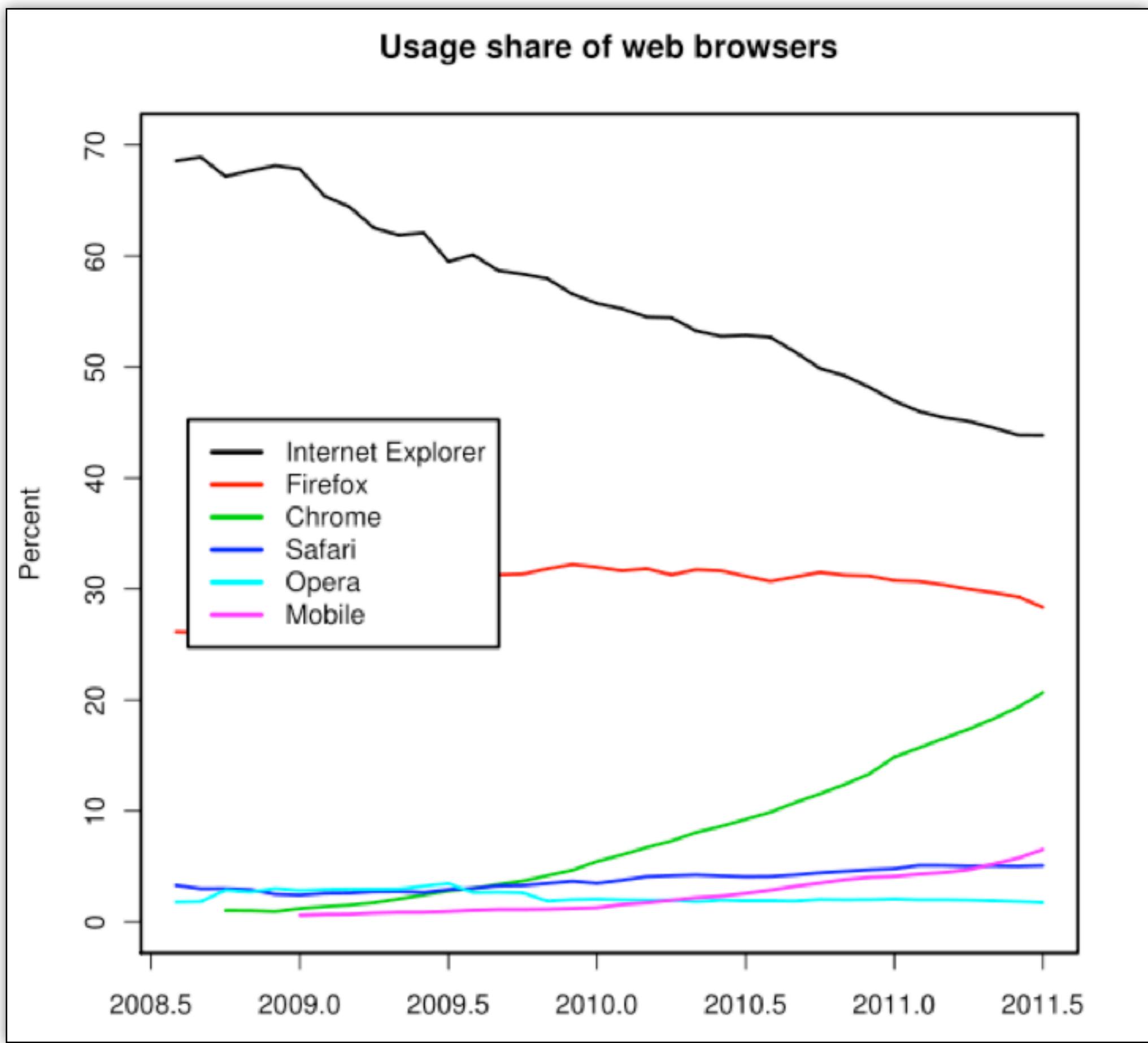
Convergence of web browser usage share (i.e. browser space is much more competitive)
- Used to be 1 primary browser, now there are 3
    - Internet Explorer
    - Google Chrome
    - Mozilla Firefox

Source:
http://en.wikipedia.org/wiki/Usage_share_of_web_browsers

modernizr2.0.4 Score Timeline

Browser using feature and innovation to compete.

Dramatic Growth of HTML5 support over last 1 1/2 years

Source:
http://www.browserscope.org/timeline?category=usertest_agt1YS1wcm9maWxlcINCxIEVGVzdBib2KQGDA

Growth of smart phones and tablets.

-Fuller browser experience
-Built with robust HTML5 / CSS3 support from beginning, and continues to have strong support (i.e. hardware acceleration of CSS3).
-Webkit based browsers dominate in this space
-Provides a solid, relatively consistent HTML5 feature set that developers can reliably target.

Is HTML5 Ready for Prime Time?

So, as we have seen, modern browsers are beginning to add support for HTML5 features, but is HTML5 ready for prime time and for production sites?
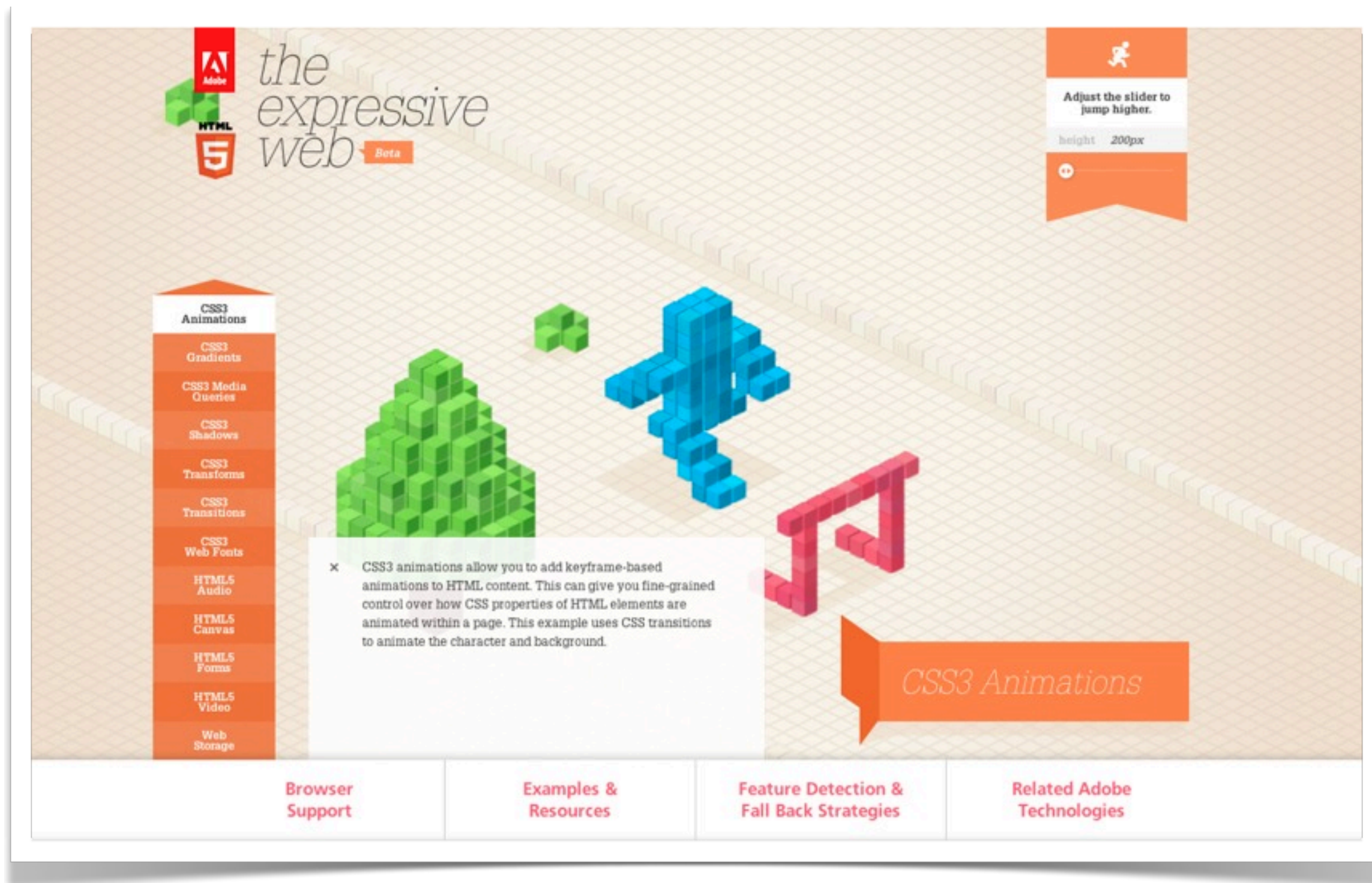
Lets look at an example that Adobe recently built.

http://beta.theexpressiveweb.com

This is a project we have put together with the goal of creating both a resource and showcase that highlights some of the most creative and expressive features being added to the web today.

More info:

http://www.adobe.com/devnet/html5/articles/introducing-expressive-web.html

What are the lessons learned?

So, what lessons regarding working with HTML5 and CSS3 did we learn from working on theexpressiveweb.com?

Think about where your site won't work

Design from the beginning with fallback in mind.

This means, figure out:

–Which browsers will you support? HTML5 enabled browsers only? or also older browsers?
–What will the experience be like when running in a browser that doesn't support some of the new features.
–What will the mobile experience be like?

Answering these questions early, will have a major impact on how you design and develop your site.

Some features are not ready for prime time

Some new HTML5 / CSS3 features have more widespread support that others.

–In general, mobile browsers will have much higher support than desktop browsers.

Minimum Browser Support: CSS3 Transforms

Internet Explorer 9.0 — Firefox 3.5 — Safari 3.1 — Chrome 1.0 — Opera 10.5 — Opera Mobile 11.0 — iOS 3.2 — Android 2.1

52.89%

Percentage of users who can view this feature (source: caniuse.com)

Minimum Browser Support: 3d Transforms

Internet Explorer NA — Firefox NA — Safari 5.0 — Chrome NA — Opera NA — Opera Mobile NA — iOS 3.2 — Android NA

4.82%

Percentage of users who can view this feature (source: caniuse.com)

Not all features have the same level of support. In general, the newer the feature, the fewer people using browsers that can use the feature.

Focus on the HTML5 / CSS3 features essential to your site

The more HTML5 / CSS3 features you add:

-The more development time implementing and debugging new features
-The more fallback cases you have to develop and implement
-The more cross browser implementation differences you have to deal with

Plan for Higher Maintenance after release

The more new HTML5 / CSS3 features you use, the more potential for features the break with browser updates.

This is particularly the case with draft, cutting edge features.

-Features not all standardized yet (still in draft)
-More differences between browser implementations
-Implementations more likely to change between releases (as feature and draft standards mature)

Performance isn't just about JavaScript

Performance isn't just about JavaScript.

Using richer, motion graphics can impact performance (or perceived performance), separate from JavaScript execution.

-Have a plan for scaling back motion graphics for platforms / browsers where they don't perform well. For example, for theexpressiveweb.com site, we don't show page transition animations on devices or older browsers.

Performance is as much about perception as raw performance.

Design and develop with mobile in mind

TheExpressiveWeb.com

–different layout for mobile and tablets
–scaled down animations to improve performance

Mobile can have dramatically different:

–Performance profiles, both for JavaScript and motion graphics
	–Sometimes better, usually worse than desktops

–Screen sizes
–Input methods

Think about how motion graphics work on a smaller screen size, and will potentially slower processors.

Desktop layouts scaled down to smaller screens rarely work well. Optimize for the device, screen and input methods.

CSS Media Queries are your friend, but also consider creating completely separate site optimized for mobile.

There is a HUGE gap between tooling and browser capabilities

This is changing, but for now, you have to write a lot of code, even for implementing motion graphics.

–Work flow and tools currently have a heavy developer focus.

–had to use custom tooling for theexpressiveweb.com site, which output code to allow developers to implement animations to site.

–Use browser debug consoles and tooling.

–Adobe Edge should address this issue (more info in later session)
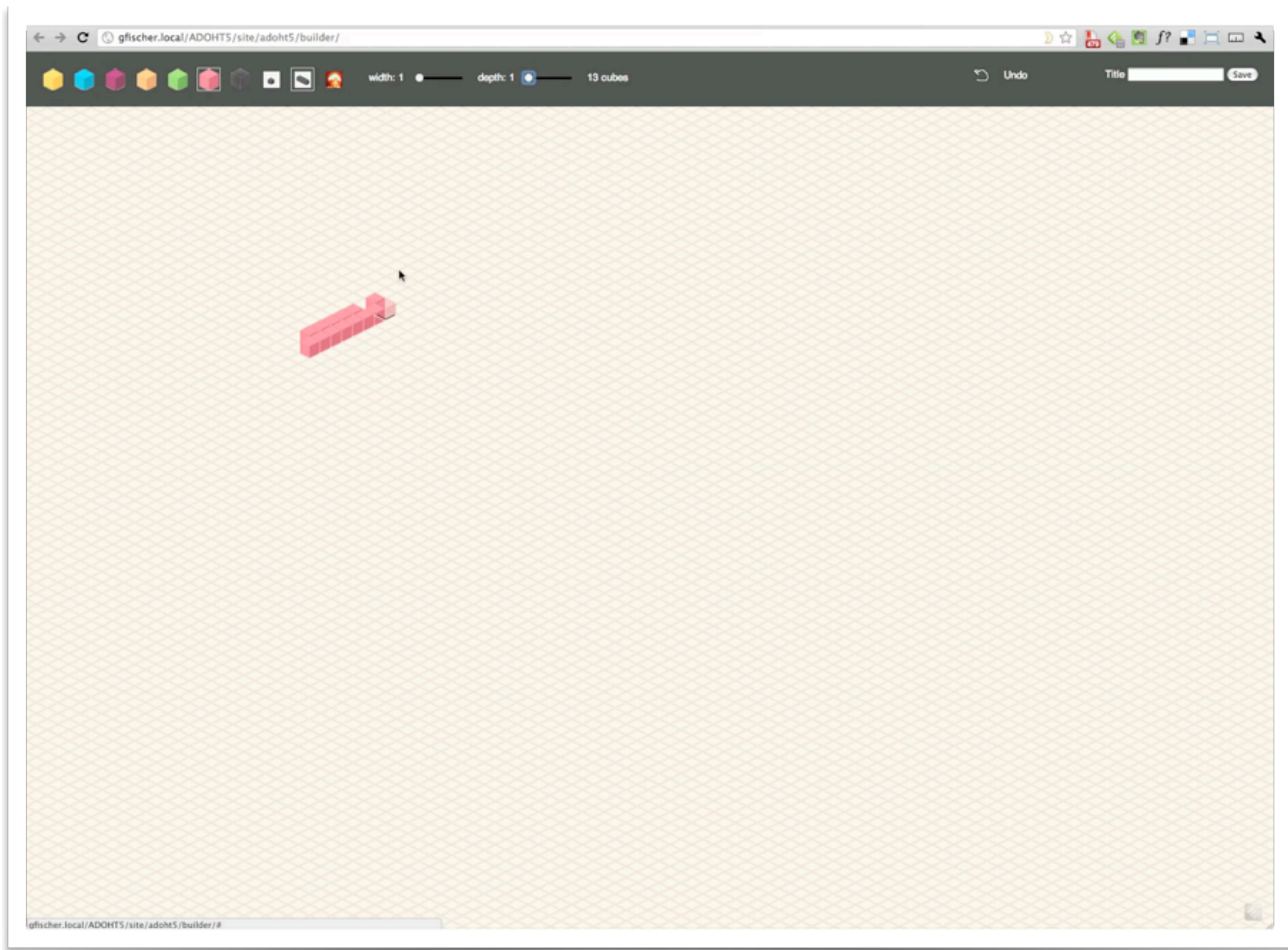
Video of custom tool created in browser to create animation for website.

Didn't use Adobe Edge preview, as we started the project while the preview was still being developed. Adobe Edge would have been perfect for laying out these animations.

Adobe and HTML5

Lets talk about the work Adobe is doing around HTML5

Tooling & Expressiveness

Adobe is committed to HTML5 and we are focusing our HTML5 efforts in two primary areas, tooling and enabling more expressive web content.

Tooling

Or course, we are continuing our leadership in HTML5 development and tooling.

Adobe has always built the top tools and technologies for Web design and development. We will continute to do that for HTML5.

In addition to our existing creative tools, which have strong support for new HTML5 and CSS3 features, we are also working on new tooling specifically focused on creating motion graphics for the browser.

Might have been surprised to see Adobe Flash listed as one of Adobe's HTML5 tools. But it is actually very relevant when talking about HTML5 authoring and tooling.

"Wallaby" is the codename for an experimental technology that converts the artwork and animation contained in Adobe Flash Professional (FLA) files into HTML. This allows you to reuse and extend the reach of your content to devices that do not support the Flash runtimes.

http://labs.adobe.com/technologies/wallaby/
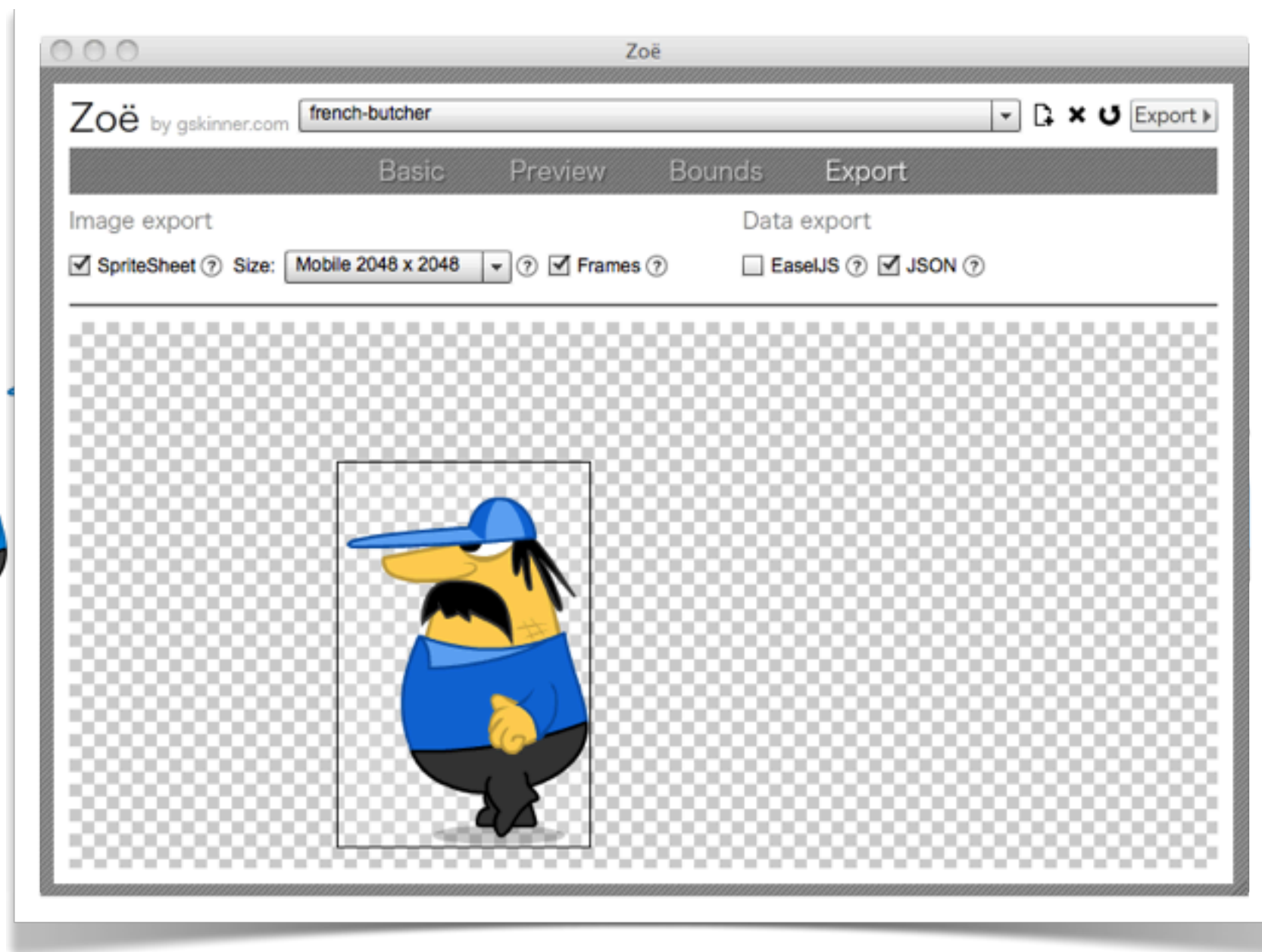
Adobe Flash is probably still the best tool for creating sprite sheet based animations for creating Canvas content.

Flash was used to create the people animations for theexpressiveweb audio page.

Community has built tooling to export Flash animations to sprite sheets.

Zoe (Grant Skinner)
http://easeljs.com/zoe.html

SWFSheet (Keith Peters)
http://www.bit-101.com/blog/?p=2939
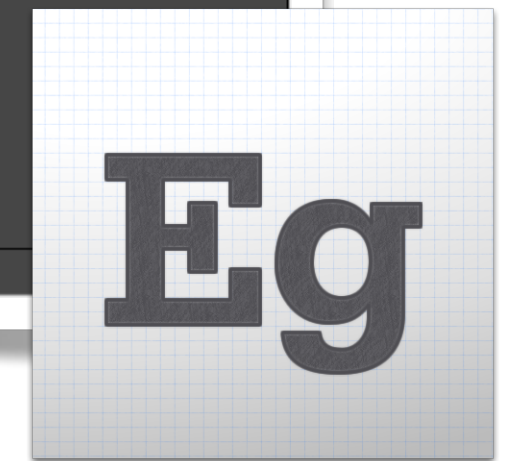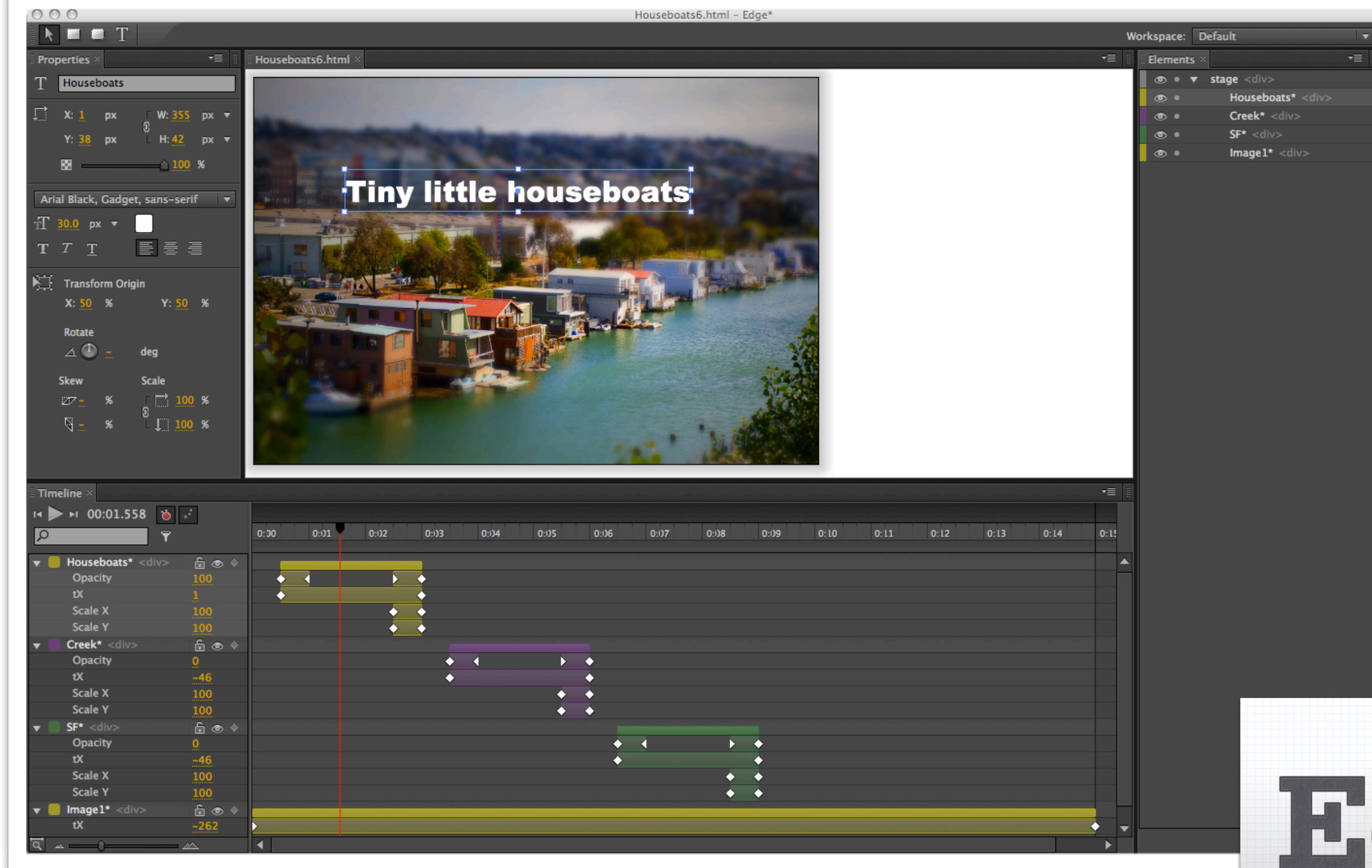
Adobe Edge is a new web motion and interaction design tool that allows designers to bring animated content to websites, using web standards like HTML5, JavaScript, and CSS3.

First public preview is available today on labs.adobe.com.

We will go into much more depth on this the Edge session later today.

Expressiveness

Adobe has a strong history of enabling expressiveness on the web (through both our tooling, and Adobe Flash). We are focusing on actively contributing to browsers and frameworks, to make it easier for designers and developers to continue to create graphically rich and expressive content for the web.
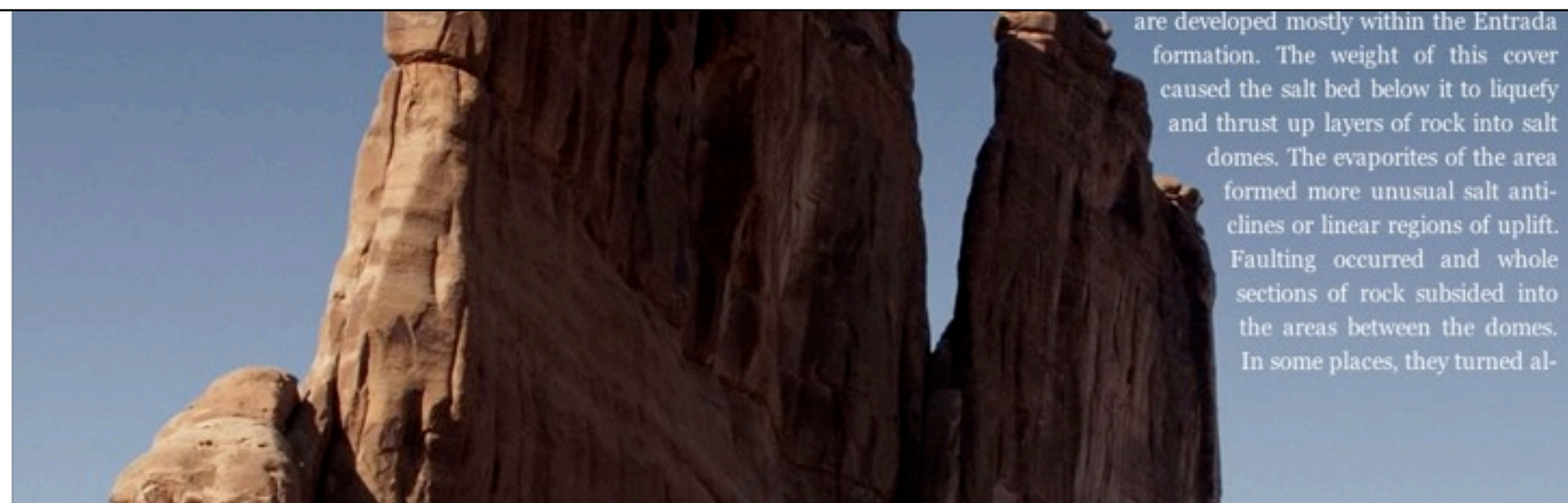
Active contributor to jQuery framework, focusing on mobile and touch input.

Dreamweaver 5.5 has support for creating mobile applications using jQuery mobile and Phonegap.
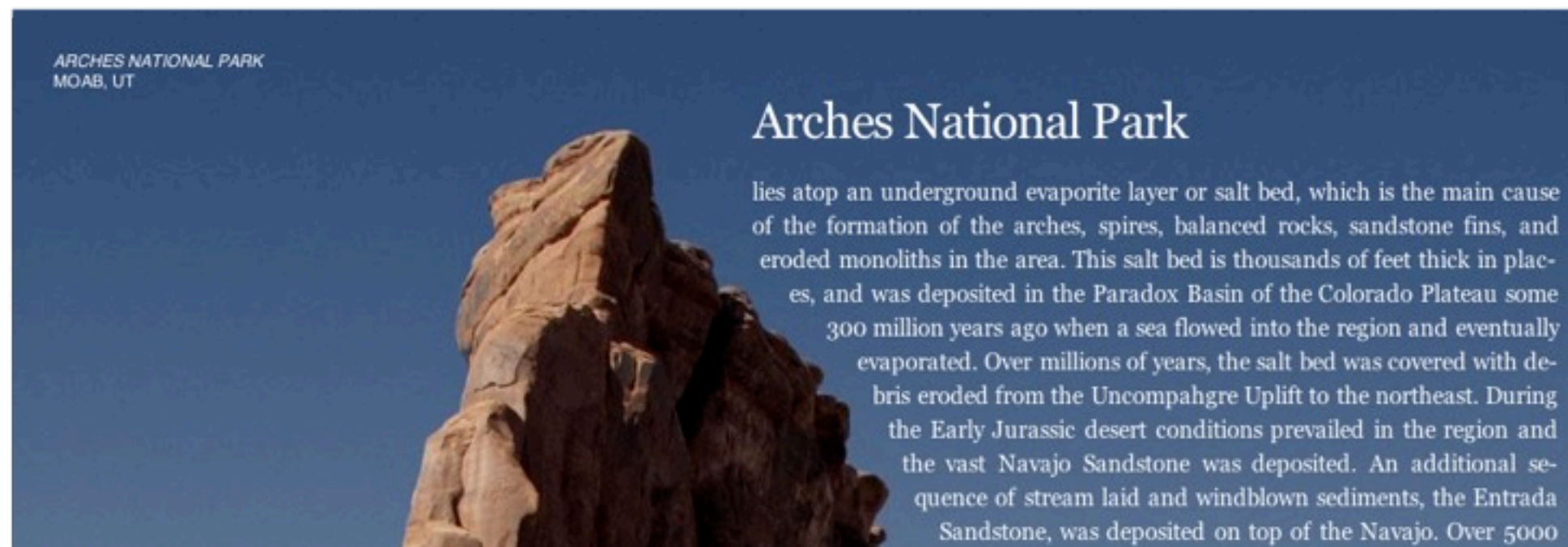
CSS Regions

We have been working on a draft proposal and implementation for making advanced text layout and reflow possible in the browser. We call this CSS regions.

Example on labs:
http://labs.adobe.com/technologies/cssregions/

Have begun contributing changes directly to webkit source.

Four many areas of functionality:
 -Threading Content : Flows from one area to another.
 -Arbitrarily Shaped Containers : Text displayed in non-rectangular areas.
 -Arbitrarily Shaped Exclusions : Text can wrap around non-rectangular areas.
 -Region Styling : Not in Prototype. Styling content depending on the area in which it flows.

More info:
http://www.adobe.com/devnet/html5/articles/css3-regions.html

Adobe Labs
labs.adobe.com

You can find more information on all of our newest technologies on:

Adobe Labs
http://labs.adobe.com

This includes:
–Adobe Edge Preview
–Wallaby
–CSS Regions

The Expressive Web
beta.theexpressiveweb.com

HTML5 Rocks
html5rocks.com

Dive into HTML5
diveintohtml5.org

Can I Use?
caniuse.com

If you want to learn more about HTML5 and CSS3, here are some great resources.

The Expressive Web Beta
http://beta.theexpressiveweb.com

HTML5 Rocks
http://html5rocks.com

Dive Into HTML5
http://www.diveintohtml5.org

Can I use?
http://www.caniuse.com

Adobe and HTML5
adobe.com/go/html5

You can always find the latest information on what Adobe is doing around HTML5 at:
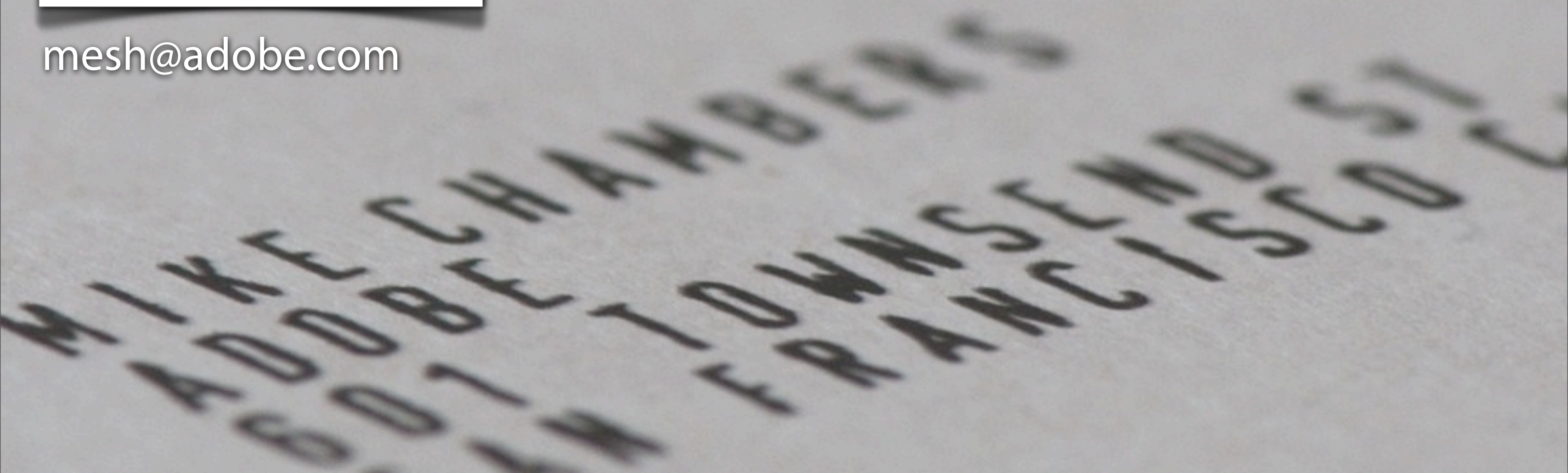
Adobe and HTML5
http://adobe.com/go/html5

Mike Chambers

mikechambers.com
github.com/mikechambers
twitter.com/mesh

mesh@adobe.com

Mike Chambers
mesh@adobe.com
http://www.mikechambers.com

Twitter : http://www.twitter.com/mesh/