



Sapient® Global Markets

23.2 68.7 40.6 56.2
+0.8 -0.9 +1.9 +1.5
42.2 23.2 56.2
-1.4 +0.8 +1.5
68.7 56.2 42.2
+0.8 -0.9 +1.4
42.2 20.4 23.2 56.2
+0.8 +1.5
23.2 68.7 56.2 42.2 40.6
+0.8 +1.9 1.5

23.2 68.7
+0.8 -0.9
42.2 23.2 56.2
-1.4 +0.8
23.2 68.7 56.2
+0.8 -0.9
42.2 20.4 23.2
23.2 40.6
+0.6



HTML5 From the Front Lines

What to embrace today (and what to avoid)

40.6 56.2
+1.5 -0.9 +1.9
68.7 42.2 20.4
+1.5 -0.9 +5.9 -1.4
68.7
-0.9 +0.8 -1.5
+1.5 -0.9 +5.9 +5.6
40.6 68.7 23.2 40.6
1.5 +1.9 +0.8

Agenda

- I. HTML5 and the Web Platform
- II. High Level HTML5 Strategy
- III. Working With HTML5 Today
- IV. Conclusion



Sapient

I. HTML5 and the Web Platform



What is HTML5?

HTML



What is HTML5?

*"HTML5 is anything you want it to be
as long as it's new and cool."*

– Peter Paul Koch



The Web Platform

- HTML5

- New Semantic elements
- Video and audio
- Form elements and input types
- Offline Web Applications

- Related Technologies

- SVG
- Canvas
- Geolocation
- Web Storage
- WebSocket

- WebGL

- EcmaScript, 5th Edition (ES5)

- CSS3

- 2D Transforms
- 3D Transforms
- CSS Animations
- CSS Transitions

- More...



Add It All Up

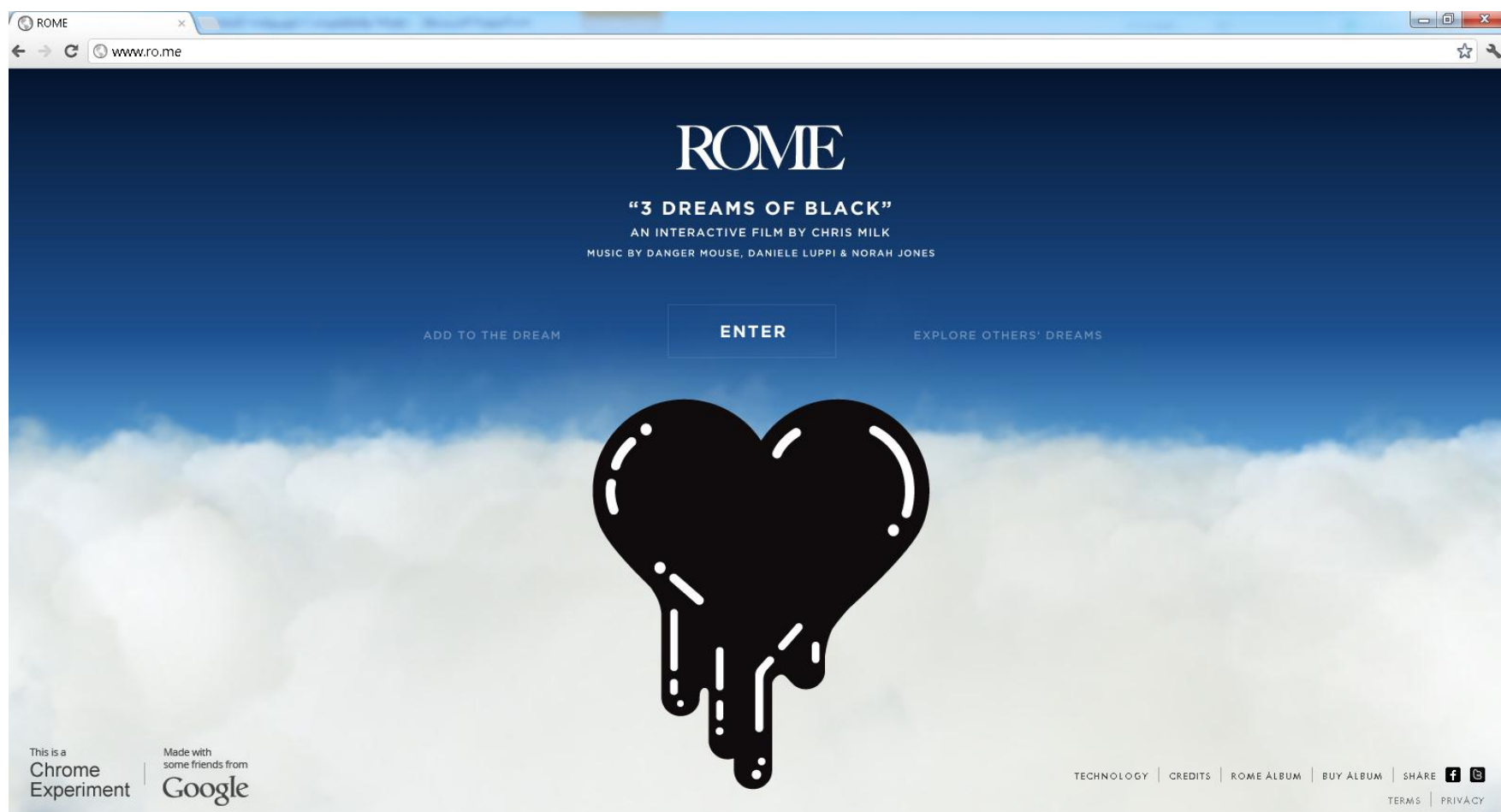




What Does That *Really* Mean?

- Much better tools for developing and deploying rich, interactive applications over the web

What Does the Future Look Like?





Sapient

II. High Level HTML5 Strategy

Set Reasonable Support Targets and Communicate

- This is as important for legacy browsers as it is for the latest/greatest desktop or mobile browsers
- On the desktop, older versions of IE will never work as well as the latest Chrome or Firefox
 - Some things just aren't going to be worth the effort for wide support. **Set expectations early**
- On Mobile **Be very specific about the devices you're going to support**
 - Plan to buy the devices
 - "Webkit" doesn't mean the same thing on every device

REMEMBER

- Your site or app doesn't have to be exactly the same in every browser
 - The only people who check a site in more than one browser are the people building it
 - Your customers will check on Desktop + mobile device(s), but they're used to different experiences on the Desktop and mobile web

Use Modernizr

<http://www.modernizr.com/>



Use Modernizr

- They say:
 - “Modernizr is the right micro-library to get you up and running with HTML5 & CSS3 today. “
 - Allows for scripting and styling of new HTML5 elements in older versions of IE
 - Tests for over 40 emerging web features
 - Creates Modernizr JS Object that contains the results of these tests as Boolean properties (e.g. `if (Modernizr.geolocation) { ... }`)
 - Adds classes to the `html` element that expose what features are and are **not supported** (e.g. `.canvas` and `.no-canvas`)
- *Doesn't actually add any functionality. No matter what the name implies.*

(Really) Use Modernizr

- Some people include Modernizr and all they **actually** use it for is supporting new HTML5 elements in old versions of Internet Explorer

- If that's all you need, use the HTML5Shiv/m:

```
/*@cc_on'abbr article aside audio canvas details figcaption figure footer header hgroup mark  
meter nav output progress section summary subline time  
video'.replace(/\\w+/g,function(n){document.createElement(n)})@*/
```

Leave the rest of the code at home

- The CSS classes and the Modernizr JavaScript object are much more important when using emerging technologies

The Final Piece: Cross Browser Polyfills

- Polyfills? Code that brings new/exciting technology to older/non-compliant browsers
- This is where the “modernizing” happens
- The people who write these are heroes
- Here’s a big list:
 - <https://github.com/Modernizr/Modernizr/wiki/HTML5-Cross-browser-Polyfills>



Sapient

From the Front Lines

Part one: Technologies in Depth

About This Section

- Introduce the feature
- Support (thanks to caniuse.com)
- Verdict & Polyfills

New Semantic Elements

- Examples: Header, Footer, Section, and Aside
 - Based on common usage patterns found during a web census conducted by editor Ian Hickson
 - `<div id="header"></div><div id="footer"></div>`, etc. becomes `<header>` and `<footer>`
- Others like `hgroup`, `mark`, `time`, and `figure` were logical additions
- Supported in all browsers except IE6,7,8- owing to the way unknown HTML elements are handled

New Semantic Elements : Verdict and Polyfills

It's worth it now. Every site I've built over the past two+ years has used the new semantic/structural elements.

Use:

- Modernizr or the HTML5 Shim(v)
- InnerShiv for dynamic content in IE
<http://jdbartlett.com/innershiv/>
- jQuery 1.7 wraps an optimized dynamic solution into the core

Canvas 2D API

- The Canvas 2D context provides a scriptable interface for drawing two-dimensional images and bitmaps in the browser
- One of the earliest stars of the HTML5 era- this got people really excited about what was possible with HTML5
- Supported in all major browsers except IE 6, 7, 8
- Great for games, charts, intense visualizations, infographics...

Canvas 2D API

- Canvas in Action

- The Wired Mind: <http://www.wired.co.uk/mind/>
- Canvas Rider <http://canvasrider.com/>
- Flotr2 <http://www.humblesoftware.com/flotr2/index>

Canvas 2D API: Verdict and Polyfills

With the understanding that there will be performance issues with more dynamic Canvas animations and games (and in older versions of IE) you can use Canvas right now.

- Use:
 - Flash Canvas <http://flashcanvas.net/>
 - supports 70% of the Canvas spec

Geolocation

- The Geolocation API is a standard interface for retrieving the geographical location of a device
- It adds a sense of place to your app
- Supported in all major browsers except IE6, 7, 8 and Safari 3.2 and 4.0

Geolocation

- Geolocation in Action

- **WikiHere**

- <http://www.somebits.com/wikihere/>

- **Twitter/Meetup Demo**

- http://experimenting.in/other/demo_geo_twitter_mashup.htm

Geolocation: Verdict and Polyfills

With design fallbacks and the available polyfill techniques geolocation can be used today

Polyfills and design aren't as good as device GPS, but they can do the job

Use:

- From WebShims
 - <https://github.com/aFarkas/webshim/blob/master/src/shims/geolocation.js>

Audio/Video

Playing audio and video in the browser is such a common event that it's easy to forget that, for most of the history of the web, there was no native method for doing so

Enter the new HTML5 audio and video elements

From a specification perspective, the inclusion of browser-native APIs for playing audio and video is straightforward. Anyone familiar with the way the replaced elements like IMG work will understand how to embed video and audio. There's a tag, a source and some attributes

With methods like `play()` and `pause()` the basics of the API are pretty easy to pick up

Audio/Video: the Code

<-- the HTML -->

```
<video src="_assets/video/sample.mp4" controls autoplay width="400" height="300" id="video-sample" data-description="sample web video"> your browser does not support the video tag
</video>
< button id="toggle"></button>
```

//the JavaScript

```
var video = document.getElementById("video-sample"),
    toggle = document.getElementById("video-toggle");
```

```
toggle.onclick = function() {
  if (video.paused) {
    video.play();
    toggle.className="playing"
  } else {
    video.pause();
    toggle.className="paused"
  }
};
```

Audio/Video: Support

- Supported in all major browsers except IE6,7,8 and Safari 3.2
- Up until March of this year, the support story was convoluted by questions of video and audio formats. This is simplified now since Mozilla has decided to support h.264 , the dominant video codec for HTML5 video.

Audio/Video: Verdict and Polyfills

People that serve video for a living have put together solutions that will work on any device seamlessly. Use them.

There are also nice solutions if you want to go it alone

- MediaElement.js <http://mediaelementjs.com/>
 - Uses <video>, flash, silverlight
- Video.js <http://videojs.com/>
 - Built on an older pattern called Video for Everybody, Video.js adds more device support with JavaScript

Audio/Video: Verdict and Polyfills

Audio follows a similar support strategy (Flash fallback)

- SoundManager2

<http://www.schillmania.com/projects/soundmanager2/>

- jPlayer <http://github.com/happyworm/jPlayer>

Web Storage

- The Web Storage specification defines an API for persistent data storage of key-value pairs in web browsers. This specification is similar to, but greatly improves upon, the functionality currently offered by cookies.
 - Cookies
 - Cookies are limited to 4k WebStorage is 500MB or more.
 - We try to avoid cookies for performance. Web storage removes the cookie overhead on every request
 - Cookies are annoying to code. I've been doing this for a long time and I only tolerate them.

Web Storage

- Storage takes two forms: `sessionStorage` and `localStorage`.
 - Each provides similar methods for managing items (`setItem()`, `removeItem()`, and `getItem()`) and for clearing the entire storage (`clear()`).
 - Session storage is designed to hold information for just the current browsing session.
 - Local storage is meant for longer-term storage of site preferences or other user data.
- Supported in all major browsers except IE6,7

Web Storage: Verdict and Polyfills

The support landscape is good and there are multiple polyfill options

storage polyfill <https://gist.github.com/350433>

Amplify.js <http://amplifyjs.com/api/store/>

PersistJS <http://pablotron.org/?cid=1557> (not a polyfill, but offers support all the way back to browsers from the 1990s.)



Sapiient

From the Front lines

Part Two: The Bonus Round



Bonus Round: WebGL

The Web-based Graphics Library (WebGL) enhances JavaScript with the ability to create interactive, three-dimensional graphics in the browser. WebGL is a context of the canvas HTML element.

(More) WebGL in Action:

chuclone <http://chuclone.com/>

Bonus Round: WebGL

- WebGL is very exciting, BUT...
- There's no hint of support in Internet Explorer
 - For older versions the polyfill path translates the 3d context of WebGL to Canvas 2D API and then would use Flash Canvas or Excanvas to render
 - There's also a commercial plugin for Internet Explorer
<http://iewebgl.com/Default.aspx>
- Even supporting browsers require up-to-date drivers and decent hardware
 - No mobile support
 - <http://blog.mozilla.com/bjacob/2011/03/28/do-users-actually-get-hardware-acceleration/>

Bonus Round: SVG

- Probably the strangest technology to be roped into the HTML5 catch-all is Scalable Vector Graphics (SVG.)
 - SVG is a vector graphics grammar defined in XML. The SVG specification has been under development by the W3C since 1999, so including it as either "new" or part of HTML5 is a stretch. But yet... people do.
- Still, newfound excitement for SVG is justified as there's now some real traction for the standard. + Raphael.js.
 - Aside: SVG Objects are DOM Objects. That makes them easier to script.
- Biggest problem right now is lack of support in Android < 2.3



Bonus Round: SVG

- SVG in Action
- HighCharts: <http://www.highcharts.com/demo/>
- D3: <http://mbostock.github.com/d3/ex/>

Bonus Round: CSS3

- A new, modular approach to CSS specifications
 - Not as fast moving as the HTML working group
 - Mature Modules
 - Selectors
 - Color
 - Backgrounds and Borders
 - Multi-column layout
 - Media Queries
 - New and In-Development Modules
 - CSS Fonts Module Level 3
 - CSS 2D Transforms Module Level 3
 - CSS 3D Transforms Module Level 3
 - CSS Animations Module Level 3 and CSS Transitions Module Level 3

Bonus Round: CSS3

- CSS3 can be used in two ways:
 - Polyfills and filters for non-compliant browsers
 - Performance cost!
 - Design graceful degradation
- You can try many CSS3 features out with <http://CSS3Please.com/>
- Also:
 - <http://www.romancortes.com/blog/pure-css-coke-can/>
 - <http://aprilzero.com/>



Sapiient

CONCLUSION

Conclusion

1. Armed with knowledge and a plan for support across devices and browser, many of these technologies can be used today
2. Make cool sites and apps
3. Have fun!

23.2	56.2	40.6	68.7	51.0	42.2	23.2	56.2	68.7	51.0	42.2	23.2	56.2	68.7	42.2	20.4	23.2	56.2	4
+0.8																		
40.6																		
-1.5																		



I've seen the
FUTURE
It's in my
BROWSER



23.2
+0.8
42.2
-1.4
23.2
+0.8
42.2

Contact



Rob Larsen

github: github.com/roblarsen

Twitter: [@robreact](https://twitter.com/robreact)

Blog @ HTML + CSS + JavaScript

<http://htmlcssjavascript.com/>



Thank You!