BlackBerry

Introducing AliceJS

Cool Visual effects in CSS

LDH (aka Laurent Hasson)

Technical Director, BlackBerry Web Platform

lhasson@rim.com

@ldhasson

Survey

- How many of you are Web developers?
- How many of you know JavaScript?
- How many of you know CSS?
  - Animations and Transitions?

BlackBerry

The Story

Building a simple game

- I wanted to do simple effects such as buttons that vibrate when clicked, or cards that flip in 3D and so on.

- I had seen those effects everywhere on the iPhone, and on platforms such as Flash and Silverlight

- I knew CSS could do it, but was surprised that all I could find really were samples with boatloads of CSS markup
  - Like 50+ lines of CSS to do a simple 3D flip effect?

R U KIDDIN' ???

```
    text-shadow: 0 3px 3px #333; margin: 50px 0 0 0; text-align: center;
}
#browser {
    margin: 0 auto; text-align: center;
    text-shadow: 0 3px 3px #333;
}
    #browser em {
    font: italic 26px Georgia, Serif; color: #666;
    }
#transformers-rule {
    width: 750px; position: relative; margin: 50px auto;
}
.container {
    width: 324px; height: 412px;
    -webkit-perspective: 1000;
}
.card {
    width: 324px; height: 412px;
    border: 8px solid #fff;
    -webkit-transform-style: preserve-3d;
    -webkit-transition: 0.5s;
}
.container:hover .card {
    -webkit-transform: rotateY(180deg);
}
.face {
    position: absolute;
    -webkit-backface-visibility: hidden;
}
.megatron {
    float: left; position: absolute; top: 30px; left: 20px;
    -webkit-transform: rotate(5deg);
}
.megatron .front {
    -webkit-transform: rotateY(180deg);
    width: 284px; height: 372px; padding: 20px;
    -webkit-transform: rotateY(180deg);
    background: #a3a3a3 url(texture.png);
}
.megatron .back h2 {
    width: 287px; height: 42px; margin: 0 auto 20px auto;
    background: url(megatron-title.png); text-indent: -9999px;
}
.megatron img {
    float: right;
}
.megatron p {
    float: left; width: 185px; margin: 0 0 20px 0;
    font-size: 17px; line-height: 28px; color: #4c4c4c;
}
.megatron dl {
    font-size: 30px; font-weight: bold;
    line-height: 40px; color: #71717;
    width: 264px; padding: 5px 10px; overflow: hidden;
    background: #cfcfcf url(gradient.png) repeat-y;
    -webkit-box-shadow: 0 1px 5px #888;
    text-shadow: 0 1px 1px #f5f5f5;
}
.megatron dl dt {
    float: left; clear: both;
}
.megatron dl dd {
    float: right;
}
.optimus {
    float: left; position: absolute; top: 120px; right: 20px;
    -webkit-transform: rotate(-3deg);
}
.optimus .front {
}
.optimus .back {
    width: 284px; height: 372px; padding: 20px;
    -webkit-transform: rotateY(180deg);
    background: #a3a3a3 url(texture.png);
}
.optimus .back h2 {
    width: 292px; height: 33px; margin: 0 auto 20px auto;
    background: url(optimus-title.png); text-indent: -9999px;
}
.optimus img {
    float: right;
}
.optimus p {
    float: left; width: 185px; margin: 0 0 27px 0;
    font-size: 17px; line-height: 28px; color: #4c4c4c;
}
.optimus dl {
    font-size: 30px; font-weight: bold;
    line-height: 40px; color: #71717;
    width: 264px; padding: 5px 10px; overflow: hidden;
    background: #cfcfcf url(gradient.png) repeat-y;
    -webkit-box-shadow: 0 1px 5px #888;
    text-shadow: 0 1px 1px #f5f5f5;
}
.optimus dl dt {
    float: left; clear: both;
}
.optimus dl dd {
    float: right;
}
```

Center up the Transformers demo

Demo headers

Add position: relative, to allow positioning of child elements

Preserve-3D ensures the content is flipped properly

Perspective adds depth to the effect

Rotate by 180 degrees on the Y axis for 0.5 seconds

Hide the opposite side of the card when flipped

Transform: rotate() adds some angles to the cards

Add some padding to the rear face and adjust the width and height to accommodate

Float the logo and text side by side

Image replacement on the H2

Font styling for the DL is set

Box-shadow and text-shadow adds subtle shading effects

Like the stats up so they appear on one line, clear the float on the next DT

Optimus card is positioned to the right

Quick fix to gracefully degrade when viewed in Firefox and later versions of Internet Explorer.

http://line25.com/articles/super-cool-css-flip-effect-with-webkit-animation

It felt like this!

It should be more like this!

http://www.snorgtees.com/friday-the-12th

Isn’t there an easier way?

- I felt like Alice In Wonderland (except not as a girl), following the white rabbit anywhere it would lead me down the hole of unexplored territory

- Can I package advanced CSS transitions and animations?

- Can I go beyond the Fade, Slide, Flip, Zoom effects that seem to have defined “advanced” Web visuals since 2007?

Everything clicked

- My colleague and co-developer of AliceJS, Jim Ing (@jim_ing) then came up with the acronym, AFTER I had settled on the name (how cool is that???)

A **L**ightweight **I**ndependent **C**SS **E**ngine

AliceJS

- A micro library (<5K minified gzipped so far for everything) to package complex CSS animations and visual effects in a single line of JavaScript

```
Alice.wobble(["divA"], parameters…}
```

- Good for simple but nice looking games, as well as for "normal" apps
  - Vibrating or flashing a button for click/touch feedback
  - Dialogs and other superimposed elements to come into the screen
  - Managing graphical elements

AliceJS

- Focus on Organics so that effects are always mysteriously slightly different
- Focus on CSS likeness to make it easy to one day perhaps integrate in the CSS standard

- Completely independent, self-contained, and easily integratable into other frameworks
- Cross Platform
- Apache 2 License

Another cool Alice

DEMOS

AliceJS home page

blackberry.github.com/Alice

Media Viewer
Peek A Boo

DETAILS

Key Defining Features

- Growing number of innovative visual effects targeting games, visual apps, as well as business apps
- Simply define your HTML elements, and AliceJS takes care of everything
- AliceJS code is just one line to apply an effect to any HTML element
- Rich parameterization allows high degree of variability
- Organics adds randomization to effects that never go stale

Ultra Simple Setup

```
<div id="ELEM1">HERE IS ONE DIV</div>
<img id="ELEM2" src="/myimg.gif">
<script src="/alice/alice.core.js"></script>
<script src="/alice/alice.wobble.js"></script>
```

- Current code APIs

```
for (var i = 0; i < TotalDivs; ++i)
    alice.wobble({id: "ELEM"+i, angle: 180,
                  duration: 3000, random: 50,
                  timing: "ease-in-out",
                  origin: "-100px -100px"
                 });
```

Current work - Core

```
[e]      _elements(e|"id"|nodeList|[e|"id"]);

{x, y}   _coords(e, "top-left|top-middle|top-right
                     |middle-left|center|middle-right
                     |bottom-left|bottom-middle
                     |bottom-right"
                    | {x: "10%"|"10px", y: "10%"|"10px"} );

// randomizes between +/-percentage
int      _randomize(val, percentage);

string _easing("linear|ease|ease-in|ease-out
                |ease-in-out|cubic-bezier(x1, y1, x2, y2)");

int      _duration("2000ms" | "2s" | 2000
                   | { val: "2000ms" | "2s" | 2000,
                       randomness: "5%" | 5 });
```

Current Work - Toss

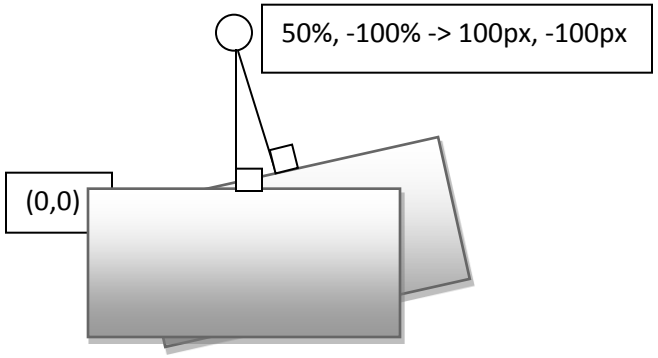
```
Toss { elements: <ELEMENTS_ARG>,
       randomness: "5%" | 5, // global randomness
       duration: <DURATION_ARGS>,
       iterations: 1 .. n | "infinite",
       timing: <TIMING_ARGS>,
       origin: { val: <ORIGIN_ARGS>,
                 spin: "right" (default) | "left",
                 randomness: “5px”, "5%" | 5
               }
       translate: { x: "0%" | "0px",
                    y: "0%" | "0px",
                    randomness: "5%" | 5
                  }
     }
```

Current Work - Rotate

```
Rotate { elements: <ELEMENT_ARGS>,
         randomness: "5%" | 5, // global randomness
         origin: { val: <ORIGIN_ARGS>,
                   randomness: “5%”
                 }
         perpendicular: true|false
         angle: { start: “0deg”,
                  end: ”180deg”|“infinite”,
                  randomness: “5%”
                },
         duration: <DURATION_ARGS>,
         iteration: { val: “infinite”|”1-n”,
                      randomness: “5%”
                    }
       }
```

Derived Effects

- Rotate provides a rich base
  - Wobble
    - Origin at the center
    - …
  - Pendulum
    - Anchor point is above, fixed and perpendicular
    - Angle is /2 from point of rest straight down
    - …
  - Pinned
    - Variation on Wobble with anchor inside element
    - …
  - …



Meant to be thought of as CSS

The syntax for Alice.js is also kept minimal and in line with CSS, which we believe to be really simple and easy to grasp. For example:

```
alice.wobble("MyDiv", 20, 5, "top-left", 1000, 10, "ease-in-out", -1);
```

This can be thought of as:

```
#MyDiv {
    animate-wobble: 20 5% top-left 1000ms 10% ease-in-out infinite;
}
```

Or more detailed:

```
#MyDiv {
  animate-wobble-rotation: 20;
  animate-wobble-rotation-ramdomness: 5%;
  animate-wobble-anchor: top-left;
  animate-wobble-duration: 1000ms;
  animate-wobble-duration-randomness: 10%;
  animate-wobble-duration-timing-function: ease-in-out;
  animate-wobble-iteration: infinite;
}
```

Core Effects

Effect	Description
Bounce	Bounce an element up and down, or sideways, with physics
Toss	Rotate and translate an element into position from any direction with over-rotation
Wobble	Rotate back and forth an element around a center point that could be inside or outside the element
Spring	Expand/contract an object with bounciness factor and physics
Rotate	Rotate a collection of elements around a center point
Flip	Flip horizontally or vertically, specifying the anchor point, through a collection of elements using a 3D view
Fade	Fade through a collection of elements

Experimental Effects

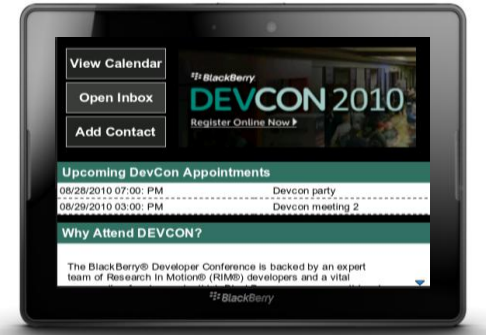
Effect	Description
Carousel	Manage multiple elements in a multi-dimentional carousel
Cube	Manage transitions through multiple elements laid out in a 3D cube horizontally or vertically.
Fold	Fold multiple elements and unfold selectively
Flip2	Explore other flip modes with transparency and other effects

Caveat

- Not sure how to clean up!!!
- When hardware accelerating a DIV, the browser creates a buffer to send to the GPU
  - See Ariya@Sencha's excellent post: http://www.sencha.com/blog/understanding-hardware-acceleration-on-mobile-browsers/
  - Basically, a 100x100 icon -> 10,000x32bit -> 40K buffer allocation that the browser may not deallocate properly
  - And when dealing with LOTs of layers promoted to hardware acceleration, may simply run out of buffer memory and crash
- Need to think more about this problem and find good heuristics

Supporting Web Developer Infrastructure

BlackBerry WebWorks



- A BlackBerry WebWorks application is created using **standard web technologies**, provides full **integration with native APIs** and has a **native application life cycle**.
- In market since 2009 on BB5. Now on BB5/6/7, and PlayBook and future QNX super phones
- BlackBerry stack for PhoneGap (which then allows you to target iOS and Android and other platforms too)
- OSS Community http://www.github.com/blackberry
- An HTML5 application doesn't mean no App store.

HTML

BlackBerry WebWorks

PhoneGap

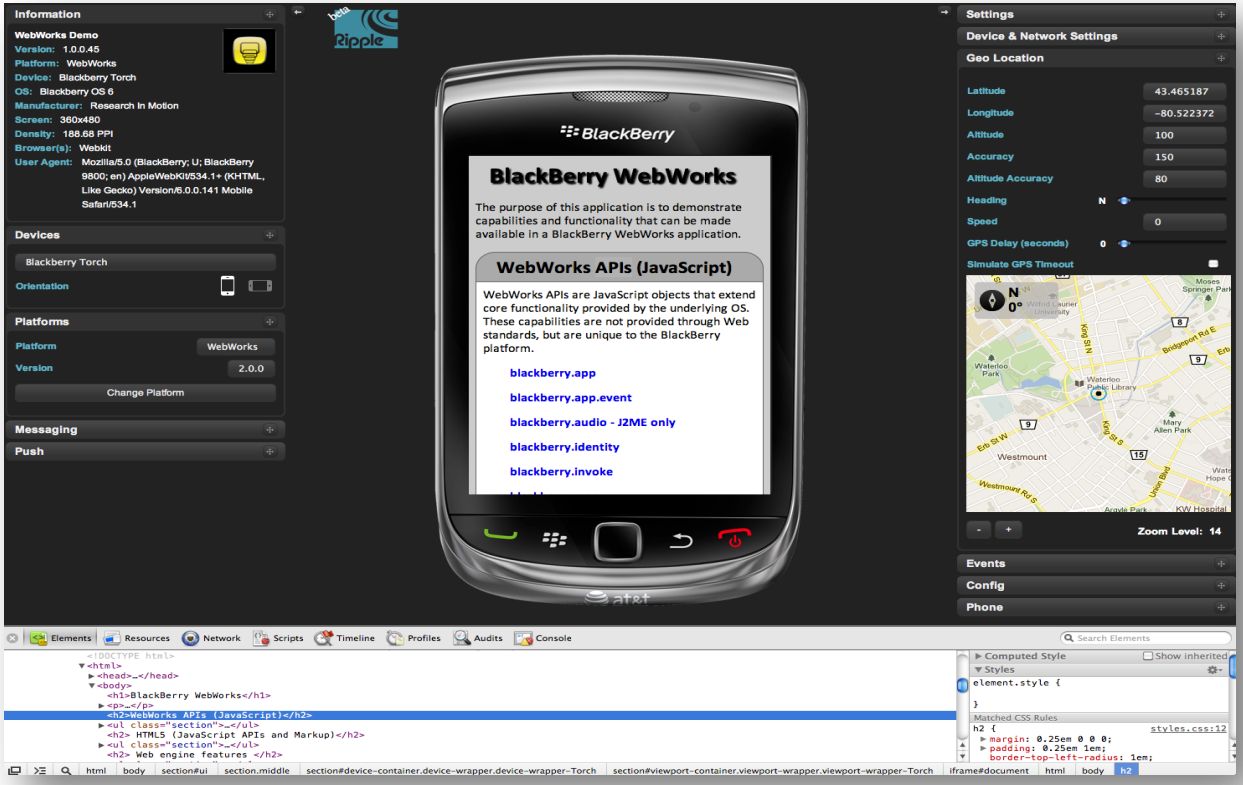
Best Web Testing

- BlackBerry Ripple allows you to do device simulation and testing right on your desktop's Google Chrome browser
  - Simulate device attribute such as screen res, skin etc…
  - Simulate device APIs and Sensors
  - Not 100% accurate, but small, fast, and conveniently browser-based

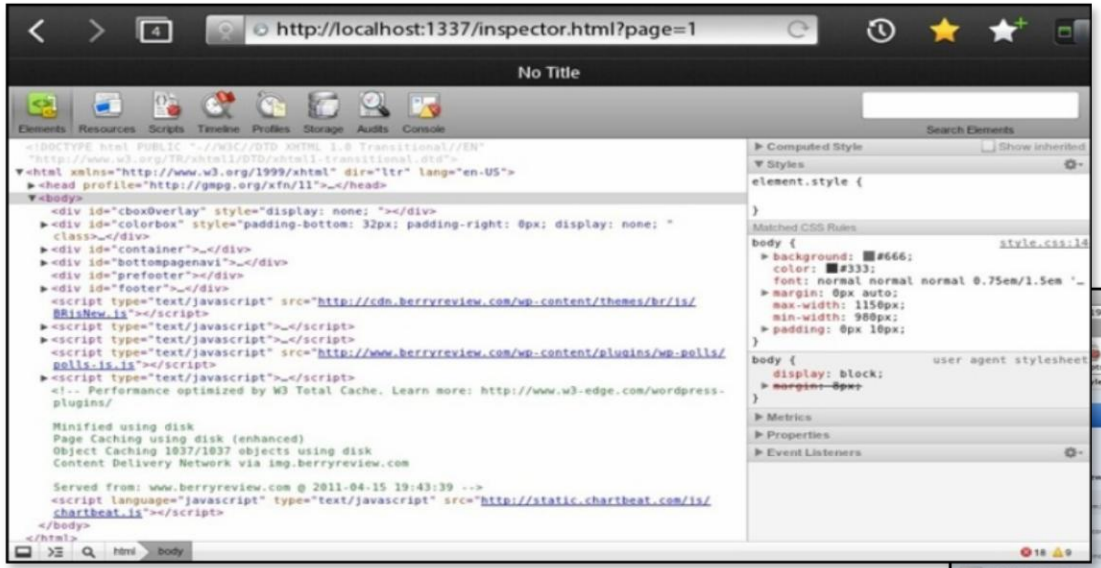
- The BlackBerry stack supports remote WebInspector directly to debug a Web app running on-device
  - You can do anything (including stepping through JS) that you can normally do in WebInspector on a desktop WebKit browser.

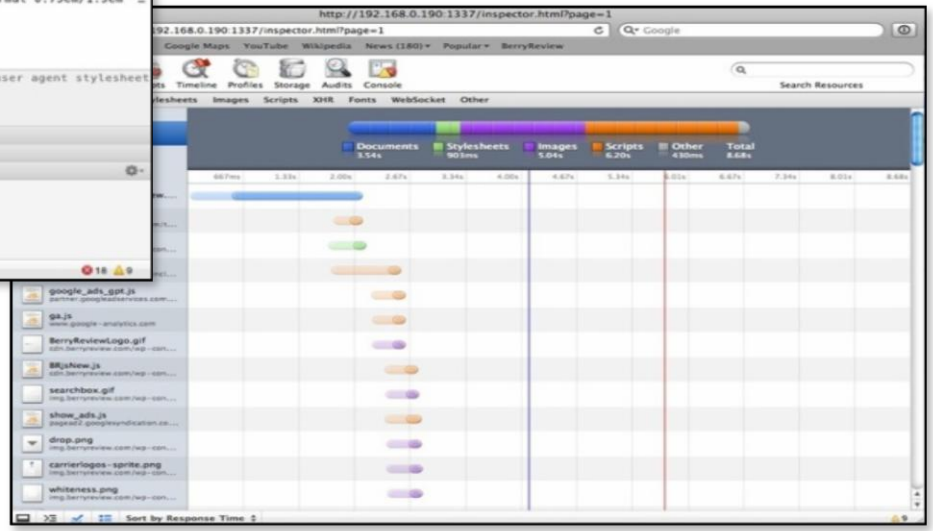
- Plus, plug you PlayBook on the USB port of your laptop, and you have bi-directional IP connectivity!!!

DEMO: Ripple


Information
WebWorks Demo
Version: 1.0.0.45
Platform: WebWorks
Device: Blackberry Torch
OS: Blackberry OS 6
Manufacturer: Research In Motion
Screen: 360x480
Density: 188.68 PPI
Browser(s): Webkit
User Agent: Mozilla/5.0 (BlackBerry; U; BlackBerry 9800; en) AppleWebKit/534.1+ (KHTML, Like Gecko) Version/6.0.0.141 Mobile Safari/534.1
Devices
Blackberry Torch
Orientation
Platforms
Platform
WebWorks
Version
2.0.0
Change Platform
Messaging
Push
Ripple
BlackBerry
BlackBerry WebWorks
The purpose of this application is to demonstrate capabilities and functionality that can be made available in a BlackBerry WebWorks application.
WebWorks APIs (JavaScript)
WebWorks APIs are JavaScript objects that extend core functionality provided by the underlying OS. These capabilities are not provided through Web standards, but are unique to the BlackBerry platform.
blackberry.app
blackberry.app.event
blackberry.audio - J2ME only
blackberry.identity
blackberry.invoke
Settings
Device & Network Settings
Geo Location
Latitude 43.465187
Longitude -80.522372
Altitude 100
Accuracy 150
Altitude Accuracy 80
Heading N
Speed 0
GPS Delay (seconds) 0
Simulate GPS Timeout
Zoom Level: 14
Events
Config
Phone
Elements
Resources
Network
Scripts
Timeline
Profiles
Audits
Console
Search Elements
<!DOCTYPE html>
<html>
<head>…</head>
<body>
<h1>BlackBerry WebWorks</h1>
…
<h2>WebWorks APIs (JavaScript)</h2>
<ul class="section">…
<h2> HTML5 (JavaScript APIs and Markup)</h2>
<ul class="section">…
<h2> Web engine features </h2>
Computed Style
Show inherited
Styles
element.style {
}
Matched CSS Rules
h2 {
styles.css:12
margin: 0.25em 0 0 0;
padding: 0.25em 1em;
border-top-left-radius: 1em;

DEMO: Web Inspector





BlackBerry

CONCLUSION

CSS3 2D and 3D animations

- AliceJS open sourced at http://blackberry.github.com/Alice

- CSS3 provides a rich visual language to do cool things, quite easily
- Performance is smooth on pretty much all modern mobile WebKit platforms
- There are stability caveats however still for the moment
- Lots of opportunities to do creative UIs, and nice simple games

Future

- Refine and simplify API
- Explore ways the app could indicate to the browser how to best manage resources
- Unit Tests
- Physics and Sensor integration for more realism
- AMD javaScript packaging
- Richer Organics
- More effects

Other “similar” frameworks

- Wink Toolkit
  - http://www.winktoolkit.org/
- SliceBox
  - http://tympanus.net/Development/Slicebox/
- FluxSlider
  - http://www.joelambert.co.uk/flux/
- Sprite3D JS
  - http://minimal.be/lab/Sprite3D/

BlackBerry Tooling URLs

- Ripple
  - http://ripple.tinyhippos.com/
- BlackBerry WebWorks
  - http://us.blackberry.com/developers/browserdev/
- BlackBerry WebWorks Handhelds Downloads
  - http://us.blackberry.com/developers/browserdev/widgetsdk.jsp
- BlackBerry WebWorks PlayBook Downloads
  - http://us.blackberry.com/developers/tablet/webworks.jsp
- WebWorks Open Source on GitHub
  - https://github.com/blackberry

THANK YOU

- Laurent Hasson (@ldhasson)