Android

It’s not (just) a phone.

demos[0] = Lamp

Who We Are

Joe McCann

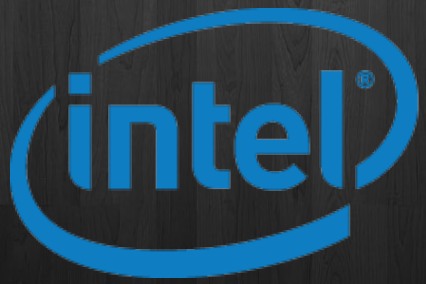
- Senior Technologist

David Wood

- Technical Architect

What We Do

Provide innovative solutions for
Fortune 500 companies



parts[0] = How we did it

Architecture

Android Client

Broadcast Server

Lamp

Architecture: Communication

- WebSocket used for communication between components
- Data passed as JSON
  - Structured as whoopingkof message

Architecture: Android Client

- Native application wrapping a WebView
  - Utilizes our Sigi framework

Architecture: Android Client

- Native application wrapping a WebView
- WebSocket implementated in Java

Architecture: Android Client

- Native application wrapping a WebView
- WebSocket implementated in Java
- WebSocket exposed to WebView through JavaScript proxy
  - Proxy provides window.WebSocket that conforms to HTML5 draft spec
  - Passes Modernizr capability detection

Architecture: Android Client

Architecture: Android Client

- Native application wrapping a WebView
- WebSocket implementated in Java
- WebSocket exposed to WebView through JavaScript proxy
- Uses our whoopingkof library for evented WebSocket communcation

Architecture: Broadcast Server

- Built on node.js
  - Express
    - http://github.com/visionmedia/express
  - node-websocket-server
    - http://github.com/miksago/node-websocket-server
  - whoopingkof library for node

Architecture: Broadcast Server

- Built on node.js
- All server code written in JavaScript

Architecture: Lamp

- Ikea Basisk lamp
  - Re-wired and fitted with RGB LED in place of light bulb
    - http://thingm.com/products/blinkm-maxm.html

Architecture: Lamp

- Ikea Basisk lamp
- Controlled by Arduino Duemilanove with an Ethernet shield

Architecture: Lamp

- Ikea Basisk lamp
- Controlled by Arduino Duemilanove with an Ethernet shield
- WebSocket client implemented using Arduino Ethernet library
  - Using aJSON library for Arduino
    - http://github.com/interactive-matter/aJson

parts[1] = Whoopingkof? Sigi?

What is whoopingkof?

- A client side JavaScript library that provides evented WebSocket communication
  - Requires use of server side whoopingkof library for node.js
  - Server side component could be easily ported to other WebSocket servers
- Usage follows jQuery event bind/unbind syntax

```
whoopingkof.bind(eventType, handler(data));
// Example:
whoopingkof.bind('chatMessage', function(data){
    alert(data.message)
});
```

What is Sigi?

- Android application framework that enables Web stack application development (à la PhoneGap)
  - Focused on rapid prototyping
- Provides mechanism for easily extending WebView
  - WebSocket implemented in WebView using this framework

parts[2] = Why Android?

Rapid Prototyping

- An extension of the design process
- Highly collaborative and iterative
- Allows for quick innovation
- Relatively inexpensive
- Allows for earlier user testing
- Enables bypassing of design artifacts
- Android openness allows for flexibility required in a rapid prototyping platform

Innovation on Android

- Innovation surface has barely been scratched due to Android's infancy in age
- Motorola CEO Sanjay Jha stated that closed phones such as Apple (iPhone) and RIM (Blackberry) work but pace of innovation in open Android is "meaningfully higher in my view"

Auxiliary Life Controller

- Capabilities of phones extend beyond making calls
- Ability to rapid prototype and innovate on Android opens possibility of an Auxiliary Life Controller (ALC)

What is an Auxiliary Life Controller?

A device to control your world in real time

parts[3] = Why the web stack?

Proving Ideas in Android Can Be Expensive

- Limited number of UI components out of the box

Proving Ideas in Android Can Be Expensive

- Limited number of UI components out of the box
- Time-consuming to build rich UIs

Proving Ideas in Android Can Be Expensive

- Limited number of UI components out of the box
- Time-consuming to build rich UIs
- Setting up/integrating with Web services adds additional complexity, scope and time

Proving Ideas in Android Can Be Expensive

- Limited number of UI components out of the box
- Time-consuming to build rich UIs
- Setting up/integrating with Web services adds additional complexity, scope and time
- Mixing Java and web stack allows for better use of developer expertise

Quick Example

- A late breaking design change requires your application to pull a JSON formatted data file from your domain

Quick Example

- A late breaking design change requires your application to pull a JSON formatted data file from your domain
- In jQuery, a GET request of JSON data is a very simple task

```
$.getJSON('http://mysite.com/data.json', function(data) {
    // Do something with your JSON data
    alert(data.win);
});
```

Proving Ideas with Web Tech Stack is Cheap and Fast

- HTML5/CSS3/JavaScript

Proving Ideas with Web Tech Stack is Cheap and Fast

- HTML5/CSS3/JavaScript
- CSS Media Queries

Proving Ideas with Web Tech Stack is Cheap and Fast

- HTML5/CSS3/JavaScript
- CSS Media Queries
- Standards-based implementation allows for better adoption across devices

Proving Ideas with Web Tech Stack is Cheap and Fast

- HTML5/CSS3/JavaScript
- CSS Media Queries
- Standards-based implementation allows for better adoption across devices
- Leverage desktop web development tools and workflow

How Fast?

- Lamp demo developed in 8 hours

How Fast?

- Lamp demo developed in 8 hours
  - WebSocket implementation for Arduino: 1 hour
  - Arduino client: 1 hour
  - Re-wiring lamp / Arduino assembly: 1 hour
  - Web client (JavaScript): 1.5 hour
  - Web client UI (HTML / CSS): 3 hours

How Fast?

- Lamp demo developed in 8 hours
- All demos developed in our free time over 2 weeks

demos[0] = Lamp++

Proving Ideas with Web Tech Stack is Cheap and Fast

- HTML5/CSS3/JavaScript
- CSS Media Queries
- Standards-based implementation allow for better adoption across devices
- Leverage desktop web development tools and workflow
- Node.js
  - Non-blocking server that provides a set of bindings to V8
  - WebOS 2.0, which runs on top of WebKit/V8, will ship with node.js

Proving Ideas with Web Tech Stack is Cheap and Fast

- HTML5/CSS3/JavaScript
- CSS Media Queries
- Standards-based implementation allow for better adoption across devices
- Node.js
  - Non-blocking server that provides a set of bindings to V8
  - WebOS 2.0, which runs on top of WebKit/V8, will ship with node.js
- WebSockets

Why WebSockets?

- Most Compelling Feature of HTML5 Spec

Why WebSockets?

- Most Compelling Feature of HTML5 Spec
- Provides near real time communication/data transfer

Why WebSockets?

- Most Compelling Feature of HTML5 Spec
- Provides near real time communication/data transfer
- Rarely hear about it from designers, marketers, “social media experts”, etc.

Why WebSockets?

- Most Compelling Feature of HTML5 Spec
- Provides near real time communication/data transfer
- Rarely hear about it from designers, marketers, “social media experts”, etc.
- Real engineers know all about it.
  - 3 out of every 4 entries in Node Knockout used WebSockets
  - Nearly all of the finalists used WebSockets (including the winner)

demos[1] = SOS

How Lightweight?

- 55 lines of client side application code (formatted for legibility)
- No application specific server side code
- Uncompressed: 1285 bytes
- Uncompressed + gzip: 484 bytes
- Compressed*: 562 bytes
- Compressed* + gzip: 318 bytes gzipped

* Compressed with Google Closure Compiler service using advanced optimizations

Production Ready

- Web software stack can be production ready, not just for prototyping
  - PhoneGap

Production Ready

- Web software stack can be production ready, not just for prototyping
  - PhoneGap
  - Appcelerator Titanium

Production Ready

- Web software stack can be production ready, not just for prototyping
  - PhoneGap
  - Appcelerator Titanium
  - Adobe AIR

What About Performance?

What About Performance?

- If user experience is not negatively impacted, nanoseconds of performance gains are irrelevant

What About Performance?

- If user experience is not negatively impacted, nanoseconds of performance gains are irrelevant
- Need to balance what lives in web stack and what lives in Java

demos[2] = Chat

Small Amounts of Data

- Impressed--

* BTW, our server (graciously provided by MediaTemple) is located in California

demos[3] = Paint

Large Amounts of Data

- Impressed++

parts[4] = Creative Business

demos[4] = Annotate

Real Business Use Case

- Annotate app is actual use case of ALC while leveraging the web stack
- The demo focuses on one small interaction in a larger application

parts[5] = Recap

More Than Just a Phone

- The possibilities with Android are much more than simply using it as a mobile phone, but as an Auxiliary Life Controller or even something better

Benefit of Rapid Prototyping

- Promotes innovation through rapid iteration
- Mixing native with web stack

Youth of Android

- We have only scratched the surface with what is capable...

Now It's Your Turn

- So now it is up to you to go take your ideas and bring them to fruition.

And now the code...

- Whoopingkof
  - http://github.com/voltron/whoopingkof
- Sigi
  - http://github.com/voltron/sigi
- Conference demos
  - http://github.com/voltron/android-only

Fork the code and get going!