



**Cross-platform apps, for mobile and desktop**

*Porting Realm of Empires to mobile HTML5*

Greg Balajewicz

@GregBala

Greg.b@BDAEntertainment.com

The background of the poster is a vibrant blue and cyan abstract design. It features various geometric shapes, including squares and rectangles, some of which contain icons like a smartphone, a game controller, and a stylized 'F2P' (Free-to-Play) logo. The overall aesthetic is futuristic and tech-oriented.

**GDC**<sup>13</sup>  
**NEXT**

NOVEMBER 5-7, 2013  
EXPO DATES: NOV 5-6  
LOS ANGELES, CA

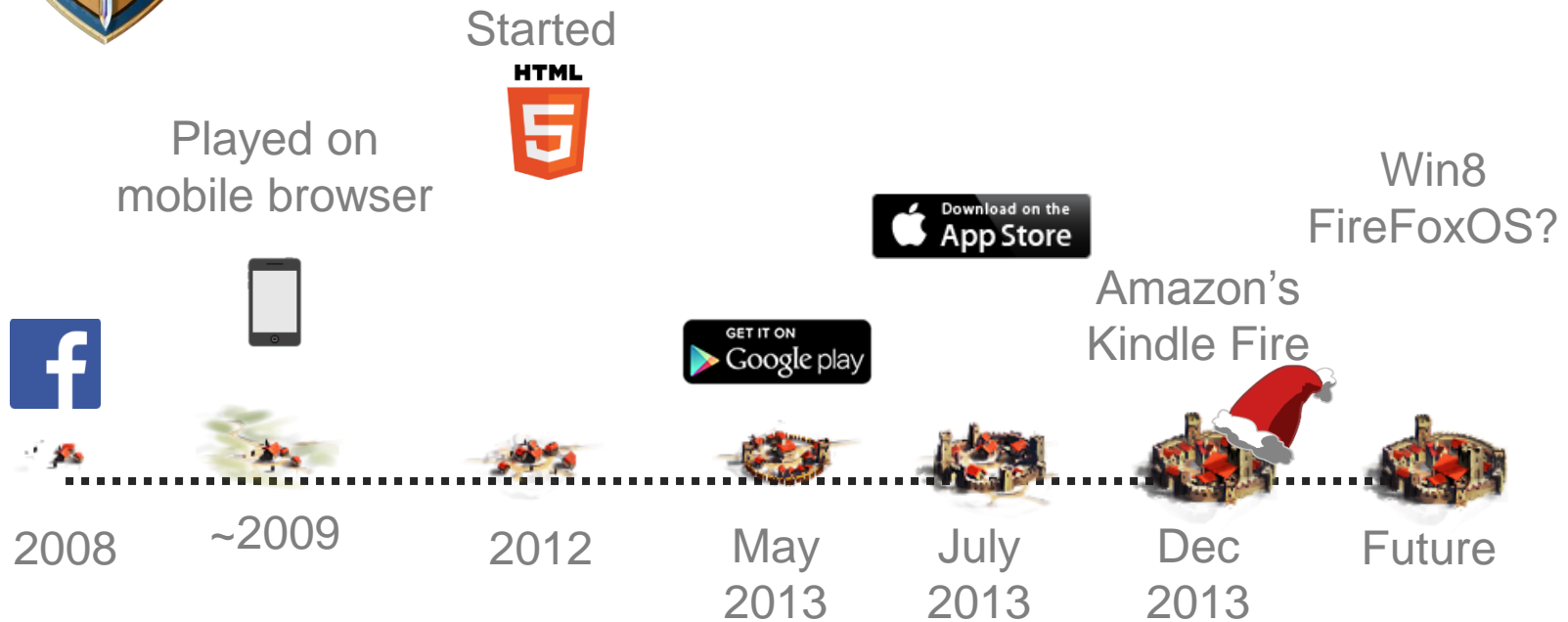
[GDCNext.com](http://GDCNext.com)



# About us



## Realm of Empires - MMORTS



# My **goal** for today

To share our experience!

Is **HTML5** right for you?



# Why **HTML5**?

Support more devices..

**easier & faster...**

not better...

**but for the future...**

Want a cross-platform solution?



...but it's not easy

# What HTML5 **is not**



# Why **HTML5** for us?

- HTML → HTML5
- Light on Animation

# Our goals


- Native-like user experience
- Run in Browser and App
- Support
  - Android v2.3+
  - iOS v5+
  - all desktop browsers
  - all other & future devices (eventually)
- Simultaneously develop for :  
Android, iOS, Desktop
- Must support Android!



# Browser **or** downloadable app?

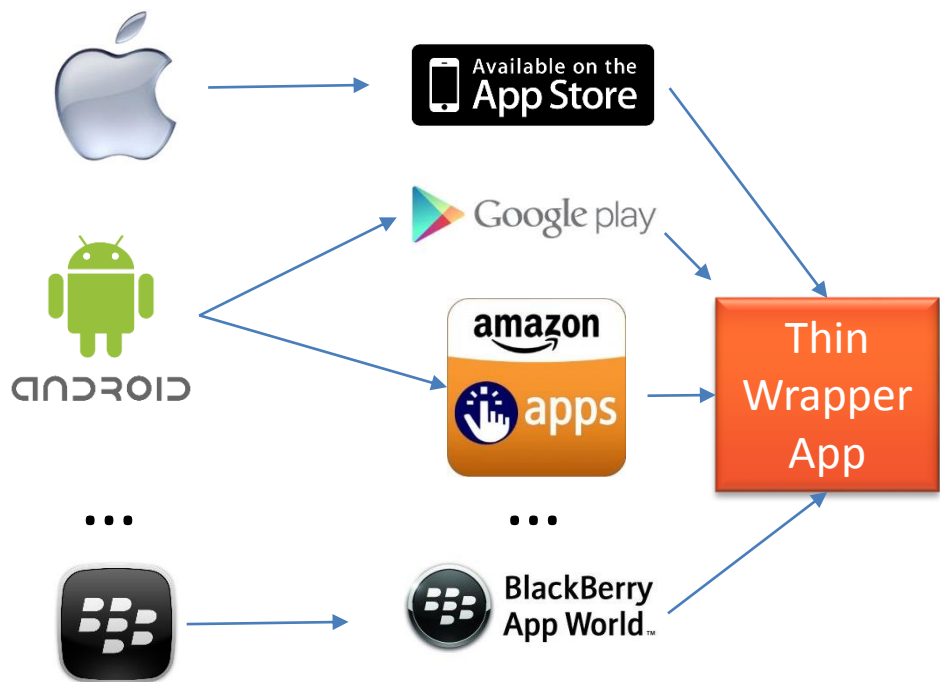
- Why App?  
Why Browser?

## Our Goals

- 
- Run in Browser and App Support
    - Android v2.3+
    - iOS v5+
    - all desktop browsers
    - all future devices (eventually)
  - Simultaneously develop for Android, iOS and desktop
  - Must support android!



# App



- ☺ discoverability
- ☺ payments
- ☺ notifications
- ☺ authentication



# Browser



- ☹ No discoverability
- ☹ No payments
- ☹ No notifications
- ☹ No authentication

➡ ☺ 5% commission

# Browser **or** downloadable app?

- No to browsers!
  - Many browsers + incompatibilities
- More benefits of app
  - Music + sounds
  - Gestures – swipe, pinch-zoom etc
- Bad decision!

## Our Goals

### ~~Run in Browser and App~~

- Support
  - Android v2.3+
  - iOS v5+
  - all desktop browsers
  - all future devices (eventually)
- Simultaneously develop for Android, iOS and desktop
- Must support android!

**GOOD NEWS!**

*You can support browsers*

# Browser **or** downloadable app?

## Lessons **Learned**

**HTML**

HTML5 *mobile* : write once, test everywhere



WebKit browsers are  
not the same on mobile



Android Stock Browser  
NOT THE SAME

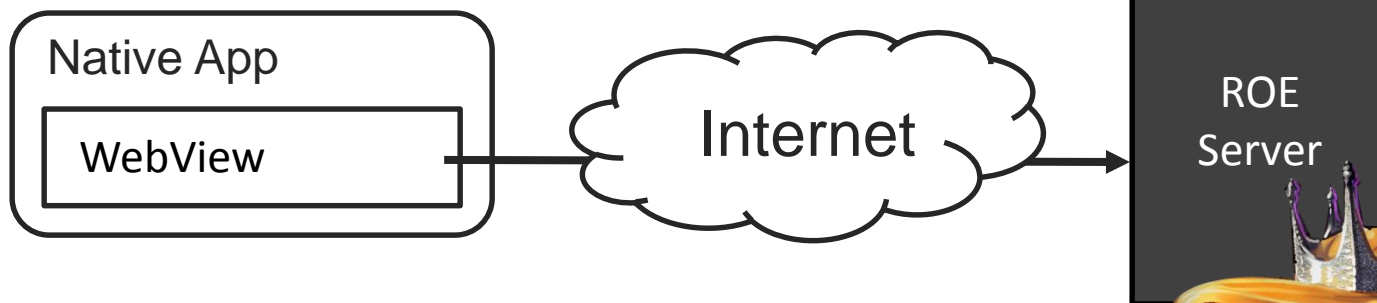
# Decision Recap

➔ Downloadable Hybrid HTML5 App



# Anatomy of the app

~~Win8~~  
~~BlackBerry~~  
~~Kindle~~  
iPhone  
Android



# Anatomy of the app

- Native
  - Java for Android
  - Objective-C for iOS
- Xamarin
  - .NET / C# => target Win8, iOS, Android
- PhoneGap
  - JavaScript => target all devices
- We did Xamarin
  - For Android
    - Due to iFrame issues, built a dual tab browser
    - Compare performance with Java – Xamarin solid
  - For iOS – iFrame issues + slow load
    - Rewrote it in Objective-C

# Anatomy of the app

## Lessons Learned

HTML5 wrapper is small - **use native**

or

**try PhoneGap** (if you don't need iFrame)



# Anatomy of the app

## Lessons Learned

Test on WebView!

Stock browser != WebView control



....



# Decision Recap

➔ Downloadable Hybrid HTML5 App

but what about the actual HTML5 part?



# HTML5 Architectural choices

Stateless,  
multi-page

Stateful,  
one-page

# HTML5 Architectural choices

Stateless,  
multi-page

Attack.php

Stateful,  
one-page

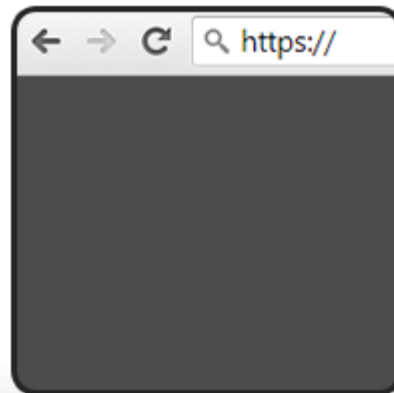
MyCity.php

Mail.php

Internet

- ✓ Easy
- ✓ Knowledgeable devs
- ✓ Short page life span

- x Slow
- x Poor UI

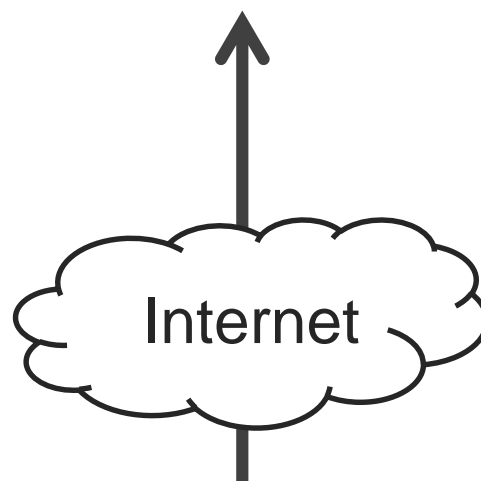


# HTML5 Architectural choices

Stateless,  
multi-page

Attack.php

Stateful,  
one-page



- ✓ Fastest UI
- ✓ Best UI

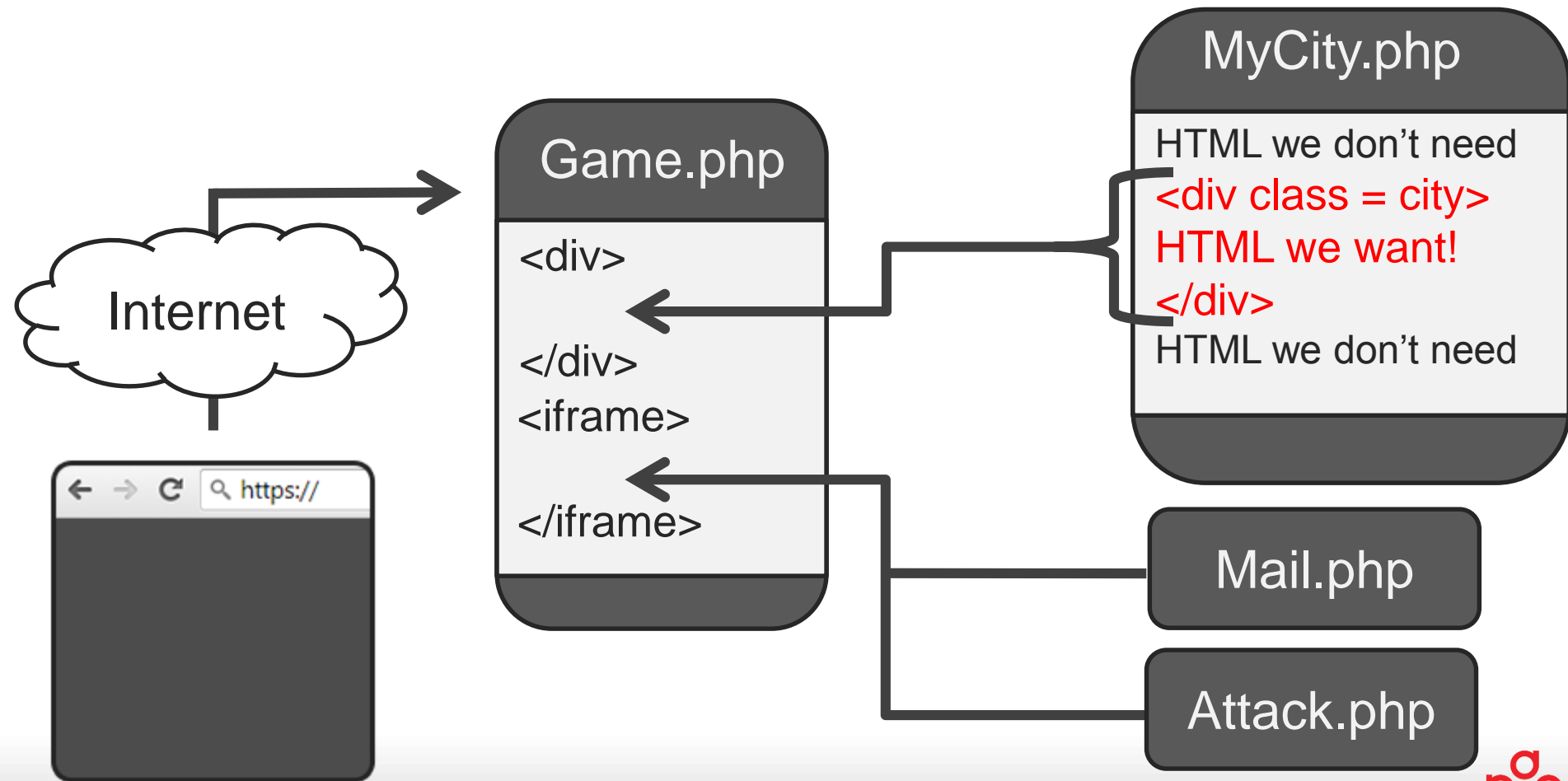
- x Difficult
- x Large app in JS
- x Lack of good devs

# HTML5 Architectural choices

Stateless,  
multi-page

Mixed /  
Stateful-Frame

Stateful,  
one-page



# HTML5 Architectural choices

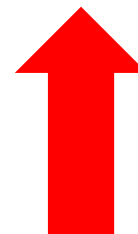
~~Stateless,  
multi-page~~

Mixed /  
Stateful-Frame

Stateful,  
one-page



Use for upgrading  
from stateless  
architecture



Use for new projects

And move to stateful





# Decision Recap

- ➔ Downloadable Hybrid HTML5 App
- ➔ Stateful-frame Architecture



# HTML5 Performance Rule of Thumb

GOOD  
NEWS!

Android  
getting  
better!

Samsung Galaxy S3  
OS v 4.1



≈

HTML5  
Performance  
wise

iPhone3GS  
iOS6

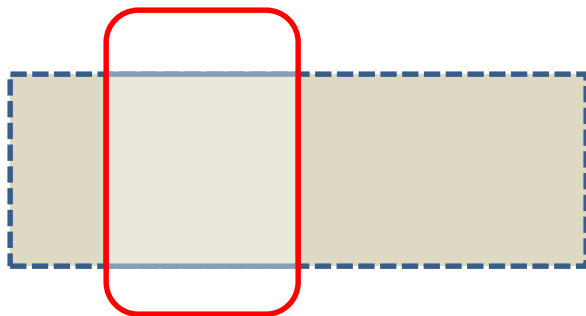


# Our findings

Only 4.0 +

No overflow handling

```
<DIV width=10>  
  <div width=1000></div>  
</DIV>
```



## Our Goals

- Native-like user experience
- Run in Browser and App
- Support
  - ~~Android v2.3+~~
  - iOS v5+
  - all desktop browsers
  - all future devices (eventually)
- Simultaneously develop for Android, iOS and desktop
- Must support android!

**GOOD NEWS!**  
< 30% android 2.3

# Our findings

Must concentrate on  
Android



## Our Goals

- Native-like user experience
- Run in ~~Browser and App~~
- Support
  - ~~Android v2.3+~~
  - iOS v5+
  - all desktop browsers
  - all future devices (eventually)
- ~~Simultaneously develop for Android, iOS and desktop~~
- Must support android!

# Our findings

## Lessons Learned

If you want to run on Android, start with Android!

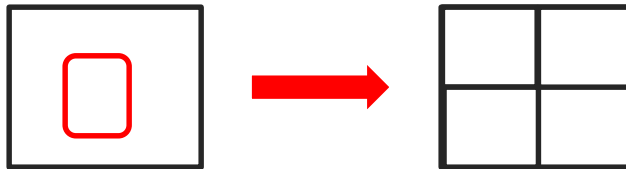


# Our findings – **performance**

- Canvas animation
  - Vector animation only on iPhone4S and up
- Sprite animation
  - 24-30fps - Galaxy S3 & iPhone 3GS
  - 2fps on many android 4.0, including tablets
- Ghost canvas bug 35474 still on Android 4.3
  - Tried DOM sprite animation – no luck

# Our findings – **performance**

- Pan vs Redraw
  - Pan large canvas
  - iOS Canvas Size Limitation





# Our findings – **performance**

- jQuery animations
  - fadeIn()
  - slideDown()

0fps – 24fps

# Our findings – performance

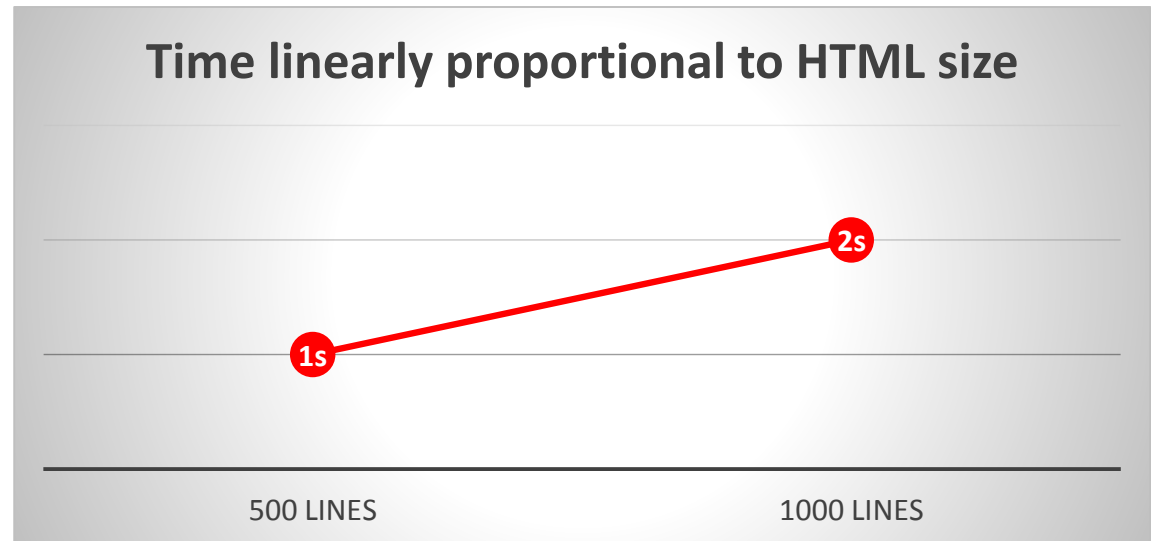
- CSS3 Transitions, animation
  - Horrible on Android 4.0
  - Android 4.1 ~ iPhone3GS
    - still choppy for non-trivial HTML
  - Improvement – don't trigger reflow

```
46 .slideLeftFrom.transition,  
47 .slideLeftTo.transition,  
48 .slideRightFrom.transition,  
49 .slideRightTo.transition {  
50     -webkit-transition: left 0.5s ease 0.0s;  
51     -moz-transition: left 0.5s ease 0.0s;  
52     -ms-transition: left 0.5s ease 0.0s;  
53     -o-transition: left 0.5s ease 0.0s;  
54     transition: left 0.5s ease 0.0s;  
55 }
```

```
46 .slideLeftFrom.transition,  
47 .slideLeftTo.transition,  
48 .slideRightFrom.transition,  
49 .slideRightTo.transition {  
50     -webkit-transition: -webkit-transform 0.5s ease 0.0s;  
51     -moz-transition: -moz-transform 0.5s ease 0.0s;  
52     -ms-transition: -ms-transform 0.5s ease 0.0s;  
53     -o-transition: -o-transform 0.5s ease 0.0s;  
54     transition: transform 0.5s ease 0.0s;  
55 }
```

# Our findings – performance

- HTML Rendering – adding HTML
  - 1 – 1.5 sec display time is typical
  - `display`: none | block
  - `visibility`: hidden | visible
  - `$.remove()` | `$.append()`



- CSS is **inconsequential**!
- Everything freezes during rendering on Android

# Our findings – performance

“click effect” - hard to do on Android

```
$('.mailButton').toggleClass('clicked-effect');  
showMailPopup();|
```

```
$('.mailButton').toggleClass('clicked-effect');  
setTimeout(showMailPopup, 300);
```

# Our findings – Data Entry

- Oh God!
- HTML5 input types? don't bet on them  
e.g. no good sliders



# Our findings – Data Entry

- Behavior HTML Structure dependent

BAD :  
display:fixed



BAD: Entry boxes in  
bottom half of screen



# Our findings – Data Entry

Get creative!









# Our findings – Data Entry

- Prototype UI early
  - on multiple devices
- Hire someone with experience
  - Expect to scrap and redo!
- Hybrid-HTML5 app: lots of options
  - WebView control
  - Native UI
  - Gestures

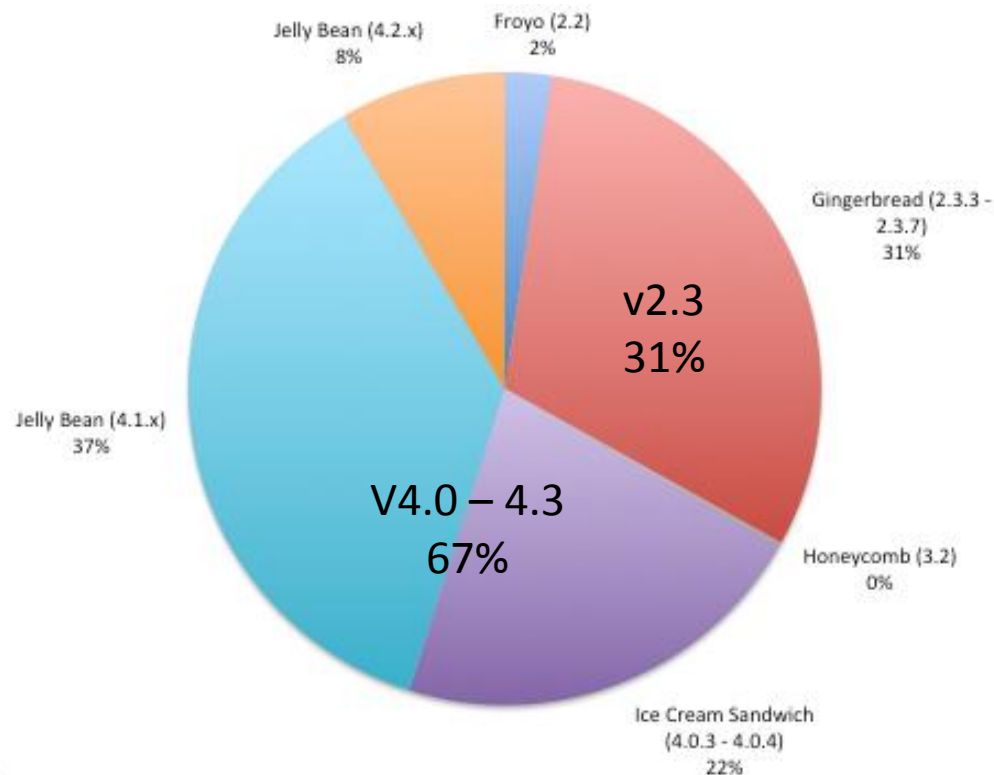
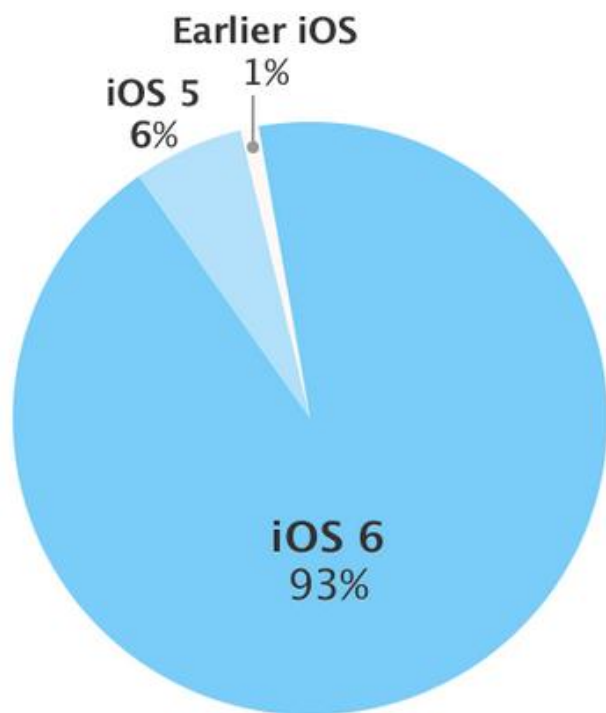


# How **native-like** can you get it?

	< 4.0	n/a
	4.0	Poor
	4.1-4.2	Ok, I guess
	4.3	Better, better
	3GS	Ok, I guess
	4S+ & iPad	Yea baby!
		Yea baby!
		Did not run

# Future of Android

Versions 4.0 – 4.3  
*for next 2 years*

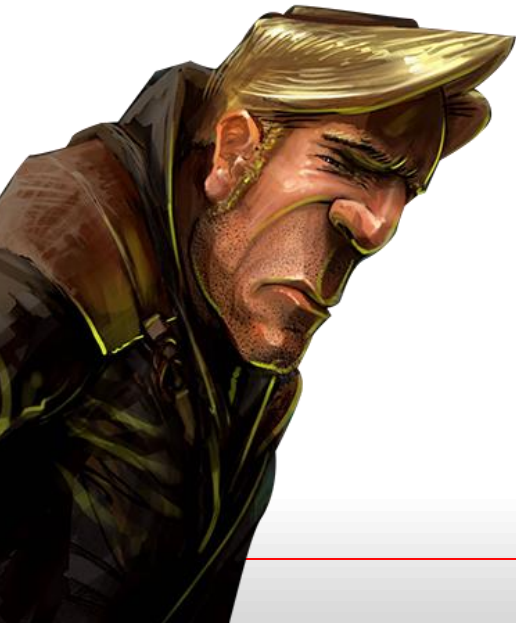


# Future of Android

- Android v4.4: Chromium WebView!
- Samsung: Android ►► Tizen



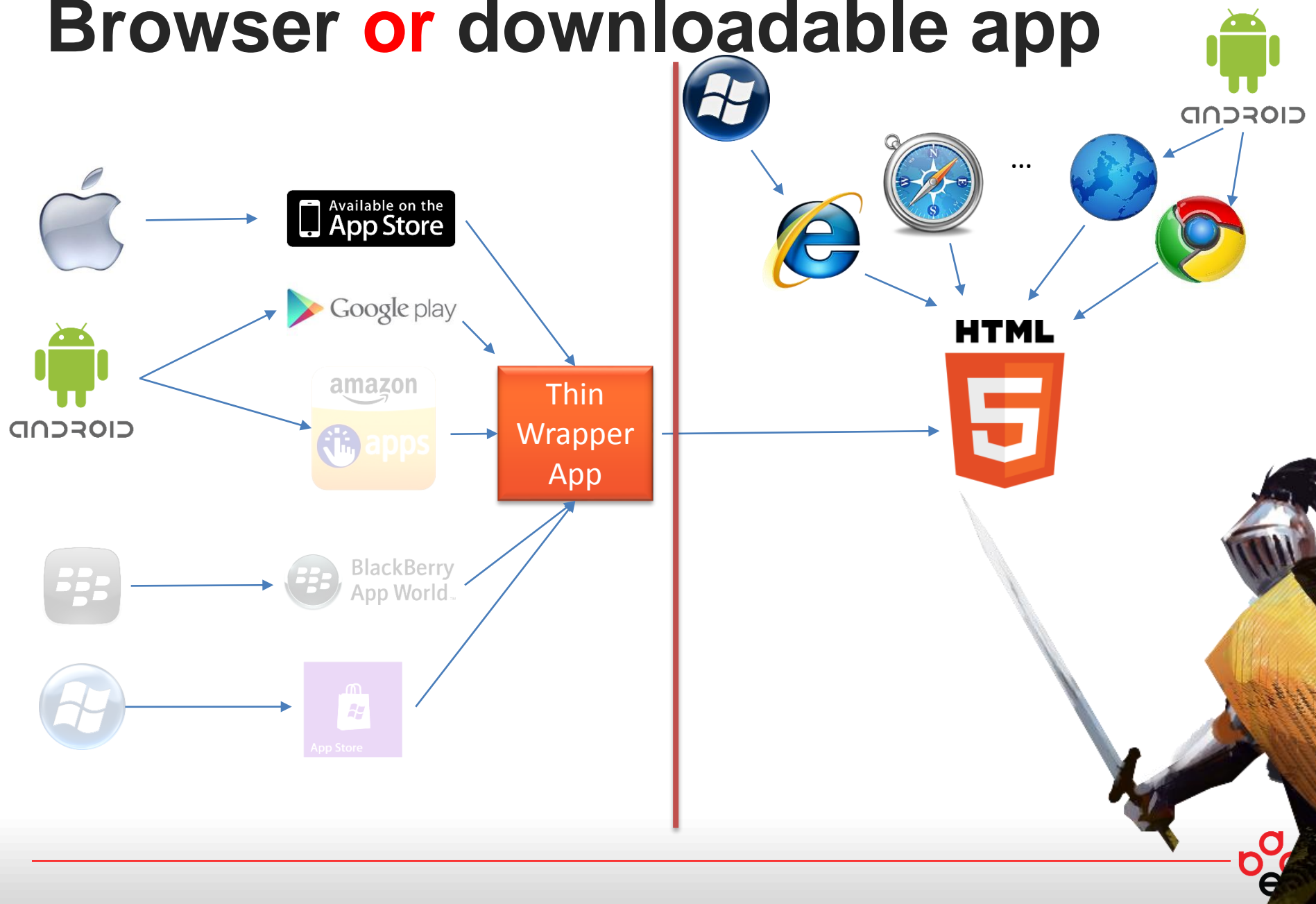
- Amazon: WebView ►► Web App



# Amazon's Web App

- Chromium based WebView!
  - ~~On all Androids v2.3+~~
  - ☹ Only on Kindle!
  - 😊 Fast
  - 😊 Authentication
  - ☹ No control over WebView
  - 😊 3 programmer in 3 weeks - go HTML5!

# Browser **or** downloadable app





# FirefoxOS

- For emerging economies
- Performance very promising



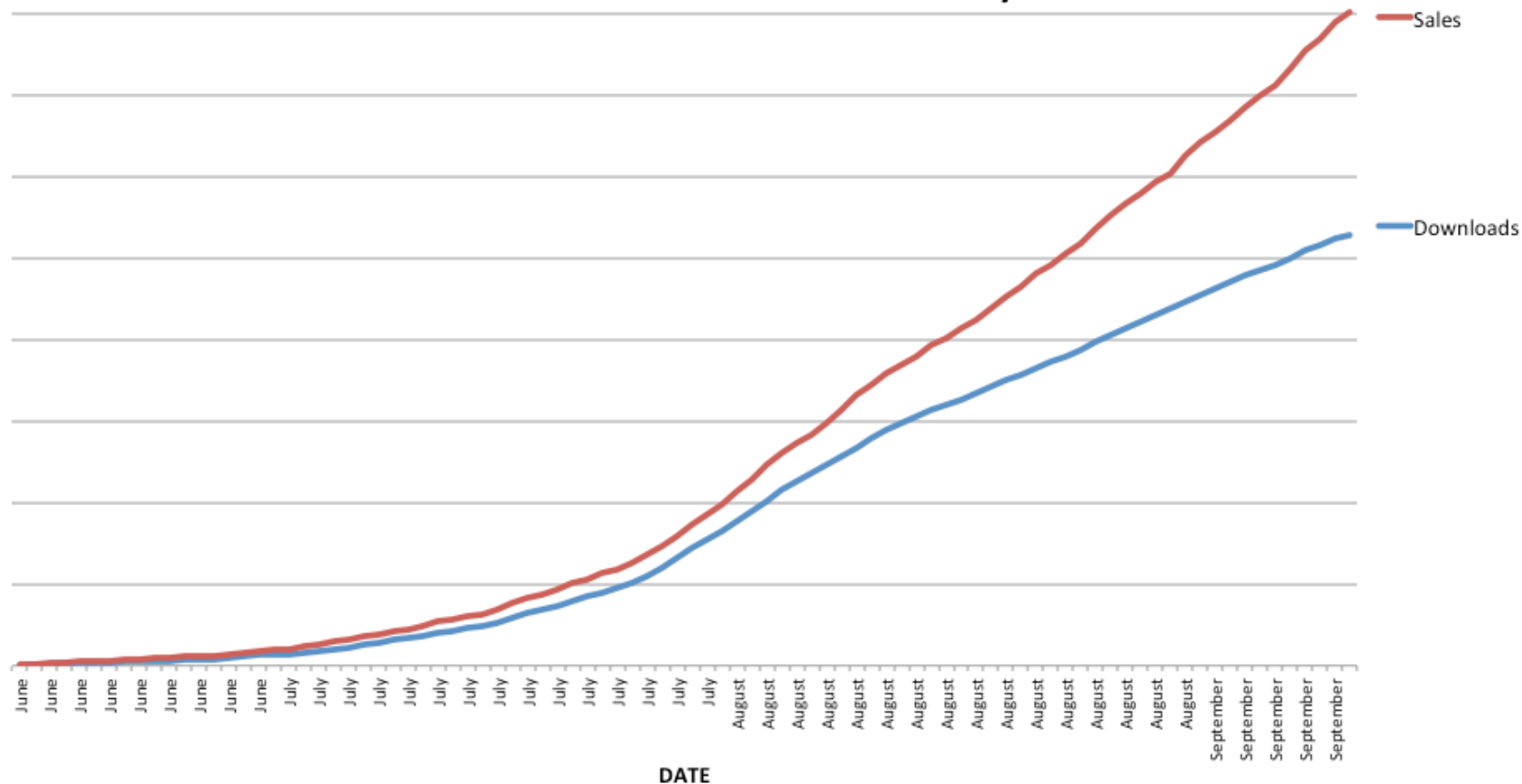


# What we would do **differently**

- Quadrupled our estimates
- Validated assumptions earlier
- Abandon HTML5?
  - HELL NO!

# You *can* make it !

RoE Mobile - Last 90 days



Questions?