WILL
HTML5 & JAVASCRIPT
SAVE THE UNIVERSE?

Trifid Nebula

http://www.nasa.gov/multimedia/imagegallery/image_feature_770.html

The Trifid Nebula, aka M20, is easy to find with a small telescope and a well-known stop in the nebula rich constellation Sagittarius. But where visible light pictures show the nebula divided into three parts by dark, obscuring dust lanes, this penetrating infrared image reveals filaments of luminous gas and newborn stars.

I work an up and coming company based in Redmond, WA. You may know us for our popular gaming console.
(Which I don't work on…)

patterns & practices

proven practices for predictable results
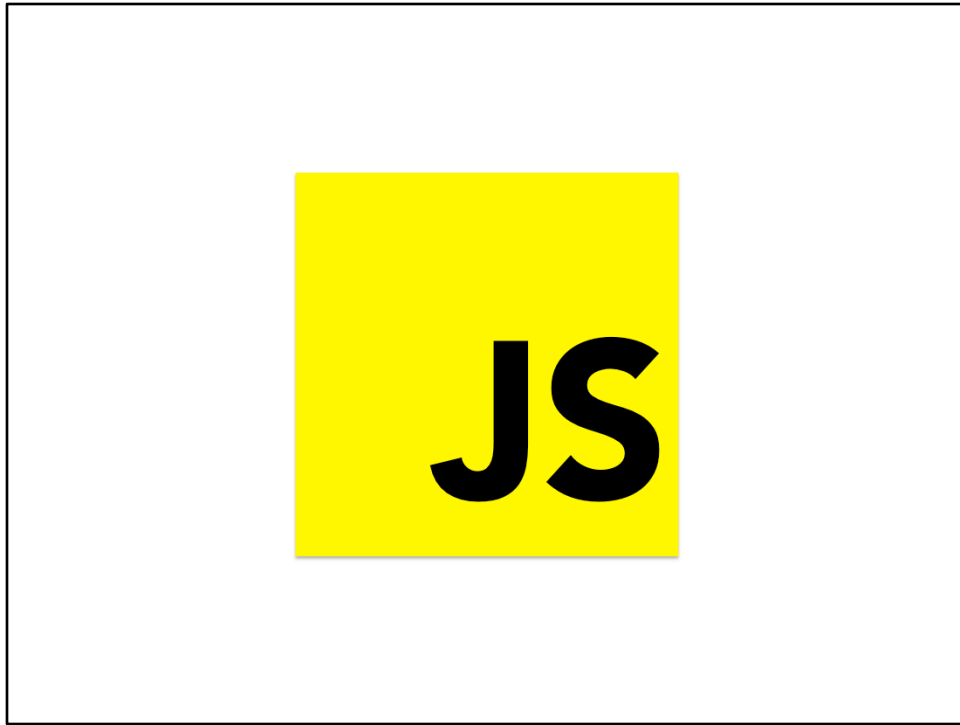
microsoft.com/practices

Even though this is my job, I'm not here as a Microsoft representative.
This talk is like twitter. The opinions expressed here are my own and do not
necessarily reflect those of my employer.

aka.ms/mobile-web

Over a year ago I had the opportunity to dive deep into the mobile web, working with industry experts like Bryan and Stephanie Rieger.

Acknowledge my bias:
I love JavaScript.

WILL
HTML5 & JAVASCRIPT
SAVE THE UNIVERSE?

This is the wrong question to ask, and I doubt that even of you are really asking this question.

What is the right question?
The purpose of this talk is ***not to give you the right answers***, but to help you ***ask the right questions***.
And there is probably more than one right question.

Is HTML5 & JavaScript

- The technology stack to bet on?
- The best path for Cross Platform development?

mu

Mu
http://en.wiktionary.org/wiki/mu#Interjection

The problem is that these questions are not specific enough. They are too general.
We cannot make such sweeping generalization about technology choices.

There's no such thing as "best practices". (You might have noticed that p&p's tag line is "proven practices".)
Ted Neward's story:
Ted "There's no such thing as best practices. Everything has a context."
Wise Guy "Yes, there is. Breathing!"
Ted "Even when you're under water?"

So, what are the right questions then?

starter questions

- What am I trying to accomplish?
- What is the value my product will bring?
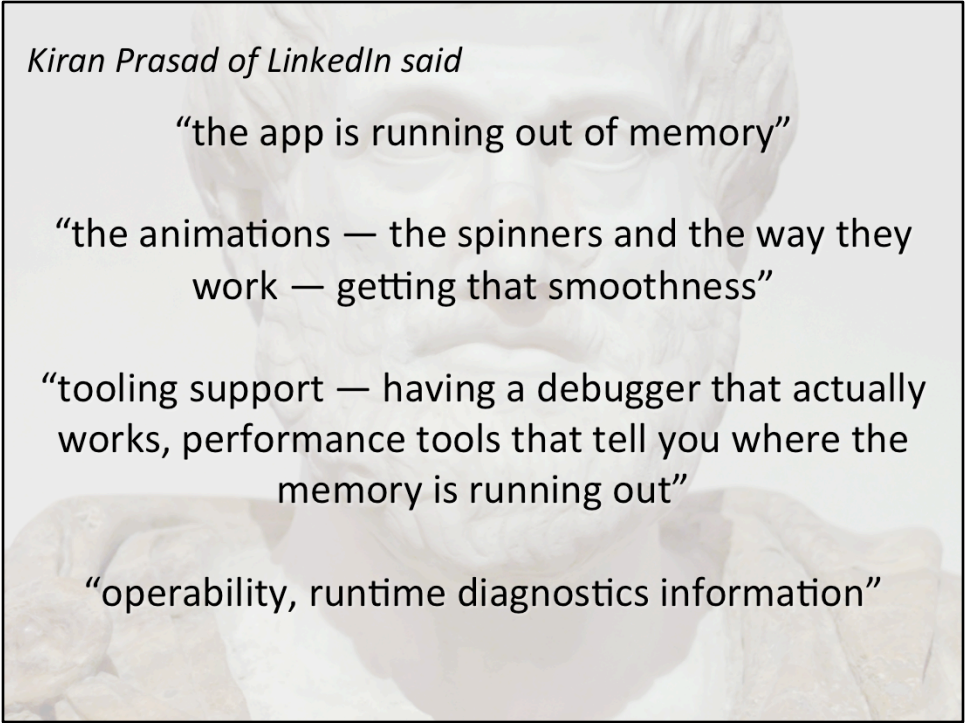- Is HTML5 & JavaScript a reasonable platform given the answers to these questions?

Here are some starter questions, in order to get us oriented in the right direction. However, even these are a bit vague. We'll get to some better ones in a moment.

facebook and LinkedIn have dumped HTML5

Image credit:
http://www.flickr.com/photos/epsos/5575089139/

Kiran Prasad of LinkedIn said

"the app is running out of memory"

"the animations — the spinners and the way they work — getting that smoothness"

"tooling support — having a debugger that actually works, performance tools that tell you where the memory is running out"

"operability, runtime diagnostics information"

Interviews with Kiran Prasad of LinkedIn (Senior Director of Engineering Mobile)

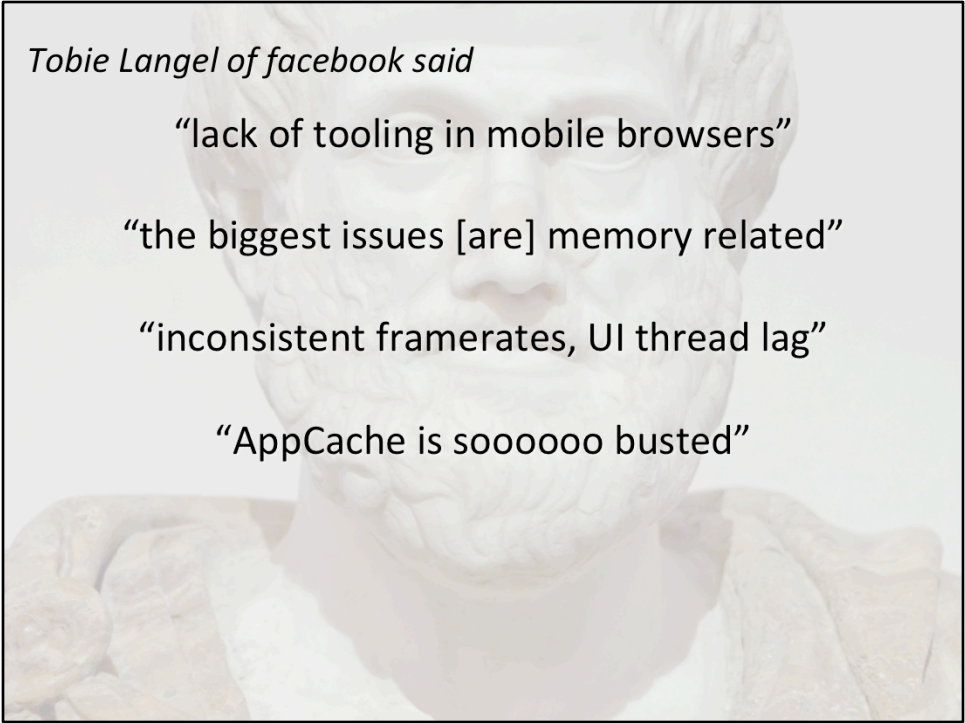**Why LinkedIn dumped HTML5 & went native for its mobile apps** (April 17, 2013)
http://venturebeat.com/2013/04/17/linkedin-mobile-web-breakup/

Not even two years after
How LinkedIn used Node.js and HTML5 to build a better, faster app (August 16, 2011)
http://venturebeat.com/2011/08/16/linkedin-node/

Image Credit:
http://en.wikipedia.org/wiki/File:Aristotle_Altemps_Inv8575.jpg

Tobie Langel of facebook said

"lack of tooling in mobile browsers"

"the biggest issues [are] memory related"

"inconsistent framerates, UI thread lag"

"AppCache is soooooo busted"

Perf Feedback - What's slowing down Mobile Facebook
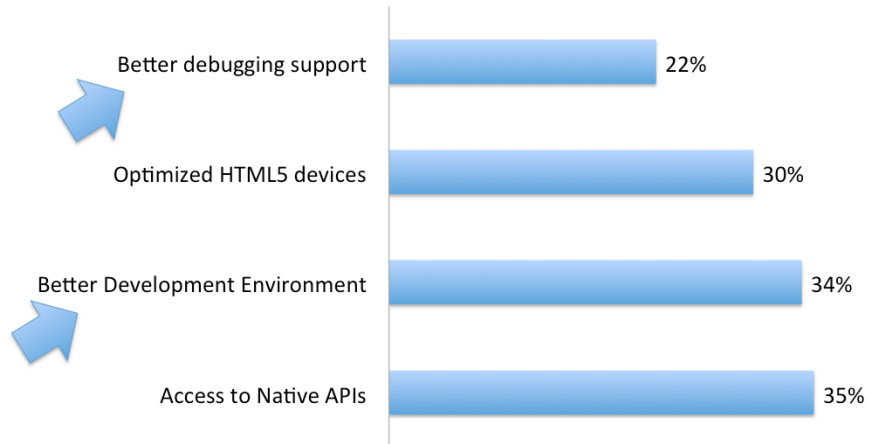http://lists.w3.org/Archives/Public/public-coremob/2012Sep/0021.html
Tobie Langel (Software Engineer at facebook, Testing Lead at W3C)

"I think the biggest mistake that we made, as a company, is betting too much on HTML5 as opposed to native"
- Mark Zuckerberg, Disrupt SF, September 2012

http://techcrunch.com/2012/09/11/watch-techcrunch-disrupt-sf-live-2/

What HTML5 needs to compete with native

Better debugging support	22%
Optimized HTML5 devices	30%
Better Development Environment	34%
Access to Native APIs	35%

VisionMobile, DeveloperEconomics.com
Survey of 3400+ developers conducted in the fall of 2012.

> "Because those two things don't exist, people are falling back to native. It's not that HTML5 isn't ready; it's that the ecosystem doesn't support it. … There are tools, but they're at the beginning. People are just figuring out the basics."
>
> *Kiran Prasad, LinkedIn*
> *April 2013*

http://venturebeat.com/2013/04/17/linkedin-mobile-web-breakup/

"…it's not that HTML5 is bad. I'm actually, on long-term, really excited about it. One of the things that's interesting is we actually have more people on a daily basis using mobile Web Facebook than we have using our iOS or Android apps combined. So mobile Web is a big thing for us."
Mark Zuckerberg, facebook
September 2012

http://lists.w3.org/Archives/Public/public-coremob/2012Sep/0015.html
Again, via Tobie Langrel on the w3 list

building apps with HTML5 is hard

A difficult fact for us enthusiasts to accept.

or at least it is hard today

a better question

What's the most
cost effective way to deliver the
best experience to users on a
diverse *set of platforms*?

Remember this talk is about Cross Platform development.

factors

What factors inform the questions?

existing skills & learning curve
Skills are portable. HTML5 is sort of like SQL, 80% of it is working on different platforms.
If this is a significant factor, you should then ask if you care about runtime portability vs development time portability?

Disruptions in technology are likely to happen, and the exact nature of those disruptions isn't something that you can necessarily predict.

If you want to reach a broad market, or be prepared for future unknown devices, then HTML5 (or just ol' old HTML4) can be a good choice.
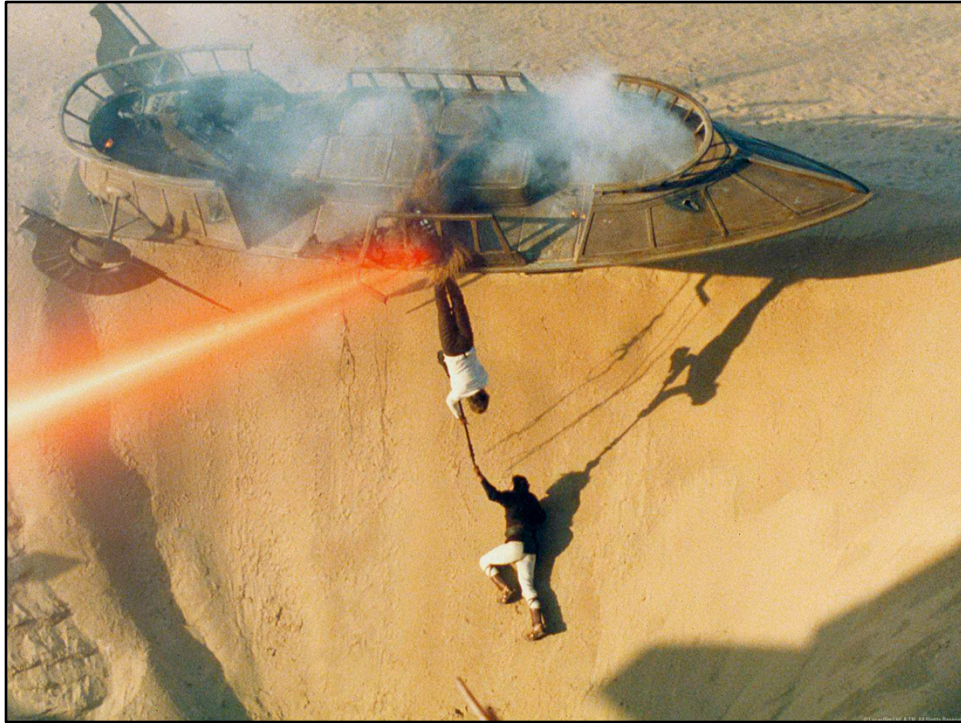
more factors

- branding & feel
- device capabilities & integration
- usage scenarios (e.g. long running apps)
- supporting your ideology

You can think of all of these factors as dimensions. You need to plot the value vs cost of choosing HTML5 along each dimension and then roll them up into a sum.
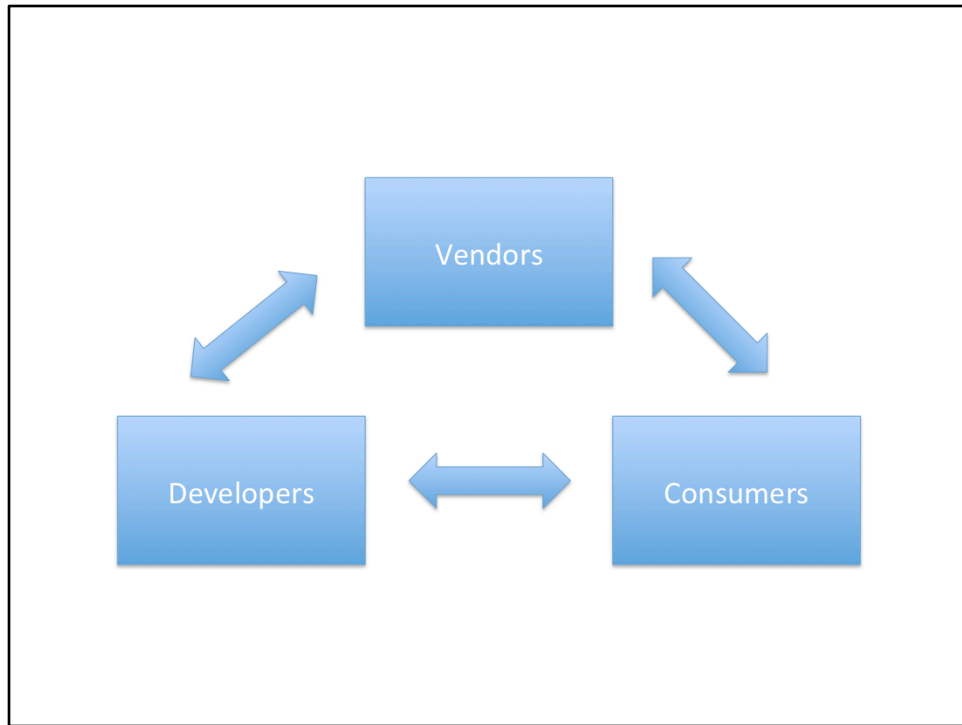
Some things are out of our hands.

There is no Pit of Success.
There are plenty of pits of failure.

be deliberate

avoid vendor specific APIs

feature, not platform detection

If you find that HTML5 and JavaScript is a the right choice, and you want to share code across platforms, then be deliberate about your design.
Avoid vendor specific APIs.
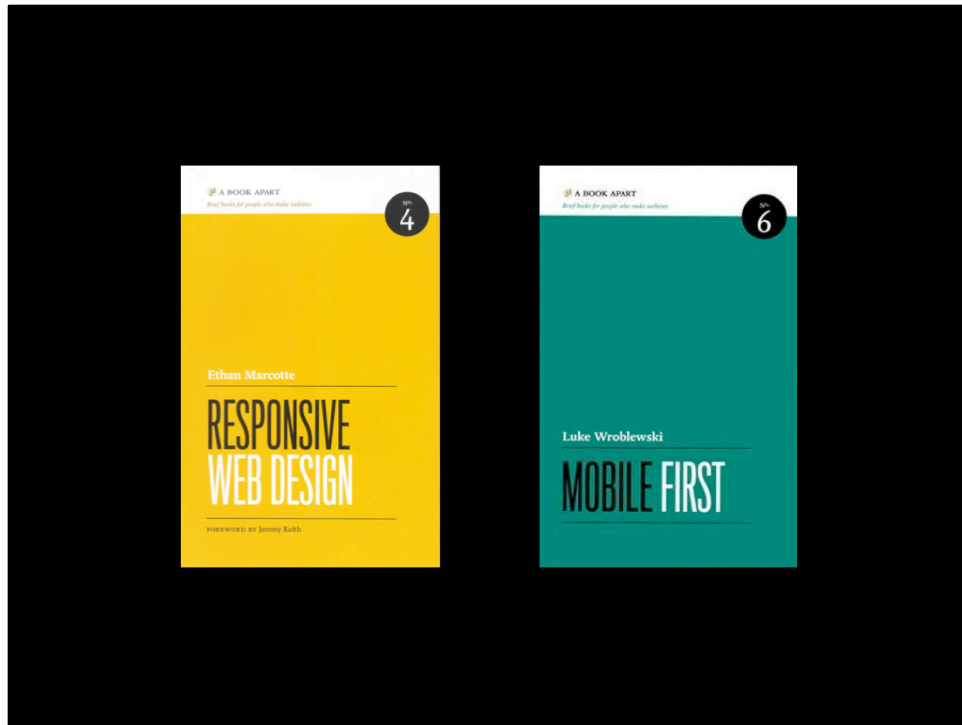Embrace feature detection, not platform/device detection.
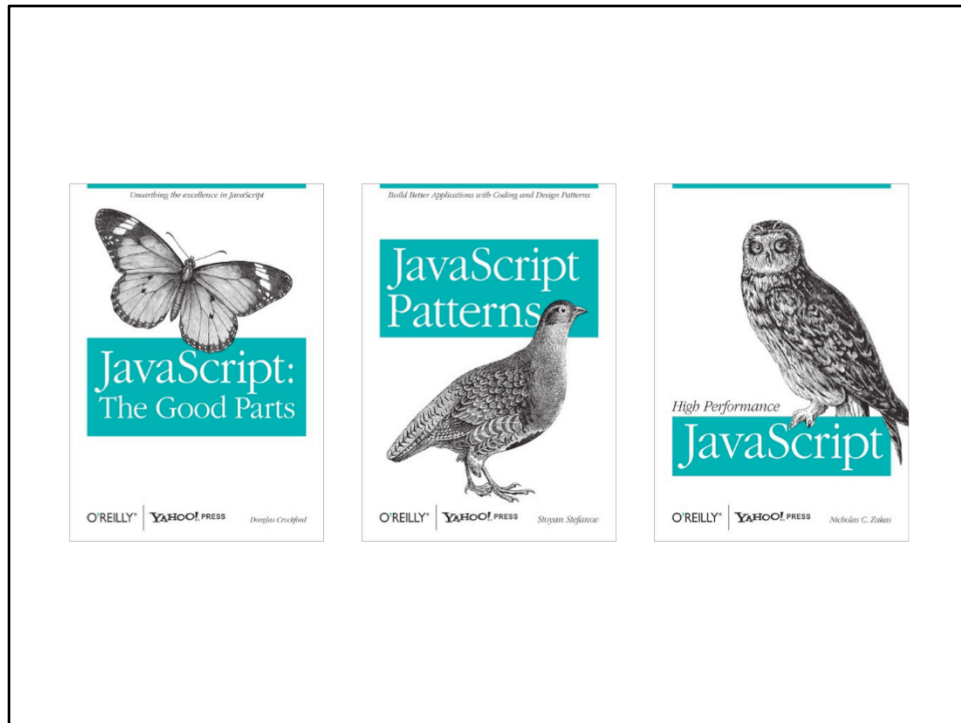
Books to consider:
- JavaScript: The Good Parts by Douglas Crockford
- JavaScript Patterns by Stoyan Stephanov
- High Performance JavaScript by Nicholas C. Zakas

Responsive Web Design by Ethan Marcotte
http://www.abookapart.com/products/responsive-web-design

Mobile First by Luke Wroblewski
http://www.abookapart.com/products/mobile-first

> "there is no single development, in either technology or management technique, which by itself promises even one order of magnitude [tenfold] improvement within a decade in productivity, in reliability, in simplicity."
>
> *Fred Brooks*
> author of The Mythical Man Month
> 1986

http://en.wikipedia.org/wiki/No_Silver_Bullet
Extracted from his paper, "No Silver Bullet — Essence and Accidents of Software Engineering" (1986).

This is somewhat misappropriating the quote, but I believe that the spirit still applies.

Is there a Myth?

The myth is that there are simple and straightforward answers that can be applied to software development.

The reality is that meaningful decisions must be made in specific contexts with specifics goals.

An animated gif of a cat dressed as shark riding a roomba chasing a duckling.

bennage

My name is Christopher Bennage, and you can find me on the interwebz.

@bennage

On twitter

dev.bennage.com

My blog

christopher bennge   aka.ms/mobile-web
@bennage
dev.bennage.com