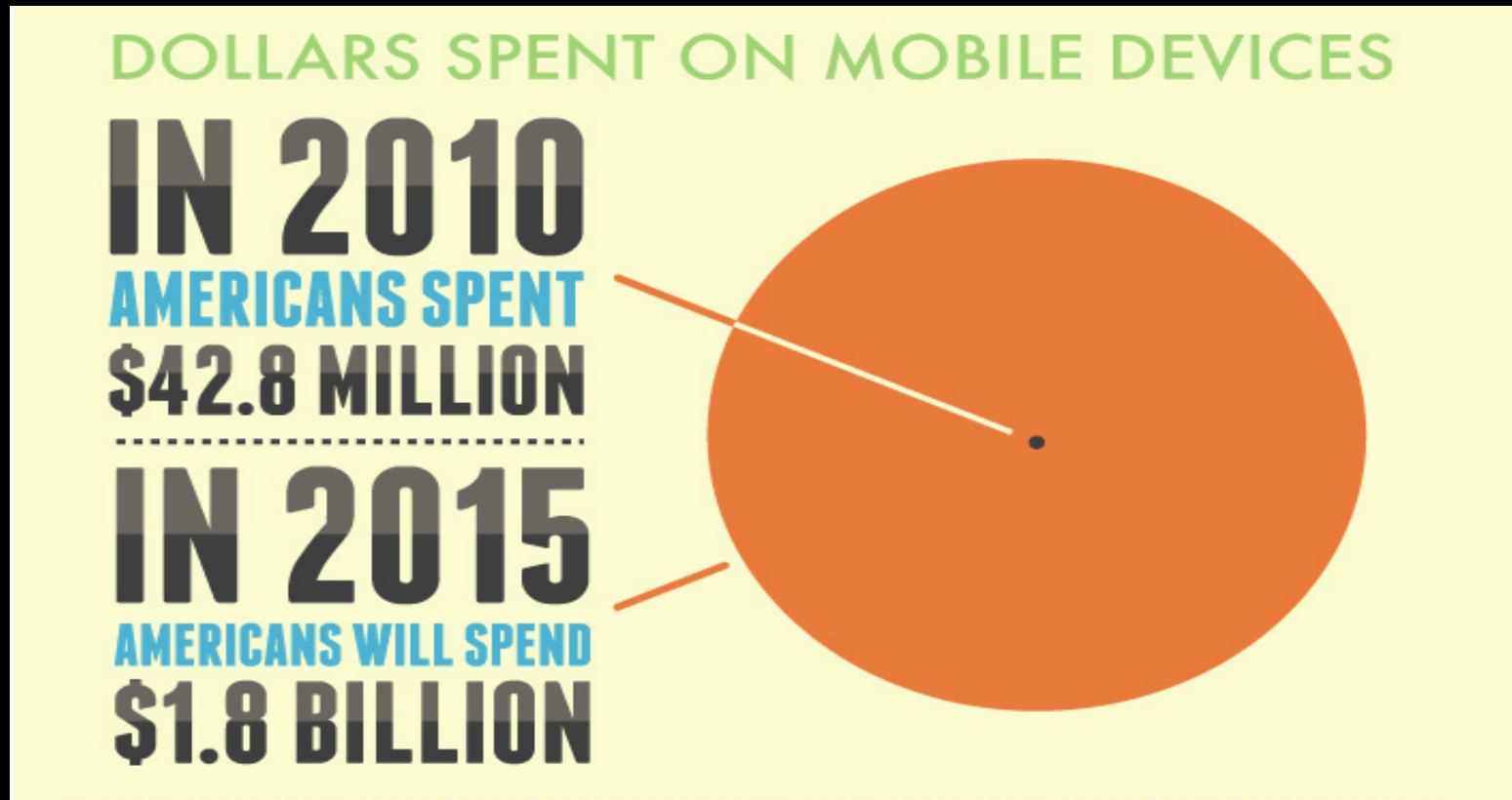# Building Native Mobile Applications with PhoneGap

Becky Gibson
Simon MacDonald

Agenda

- Mobile Application Overview
- What is PhoneGap?
- PhoneGap Architecture
- Extending PhoneGap
- Building an App
- Summary

Mobile Space

DOLLARS SPENT ON MOBILE DEVICES

IN 2010
AMERICANS SPENT
$42.8 MILLION

IN 2015
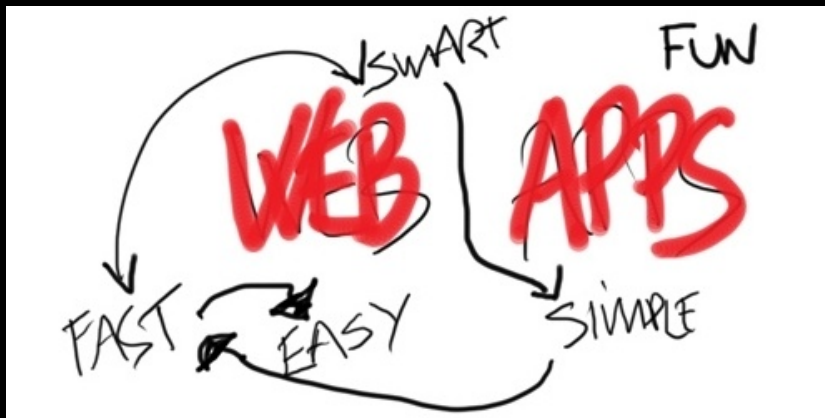AMERICANS WILL SPEND
$1.8 BILLION

http://www.abmuku.com/2011/02/24/technology/american-cell-phone-usage-infographic/

Everyone Wants a Piece of that Pie!

Mobile Web Apps

- Fast, easy, simple
- Run in mobile browsers
- Built on W3C technologies
- Minimal access to device features

Mobile Applications

- Platform Specific
- Sold via App Stores
- Specific technologies and development environments
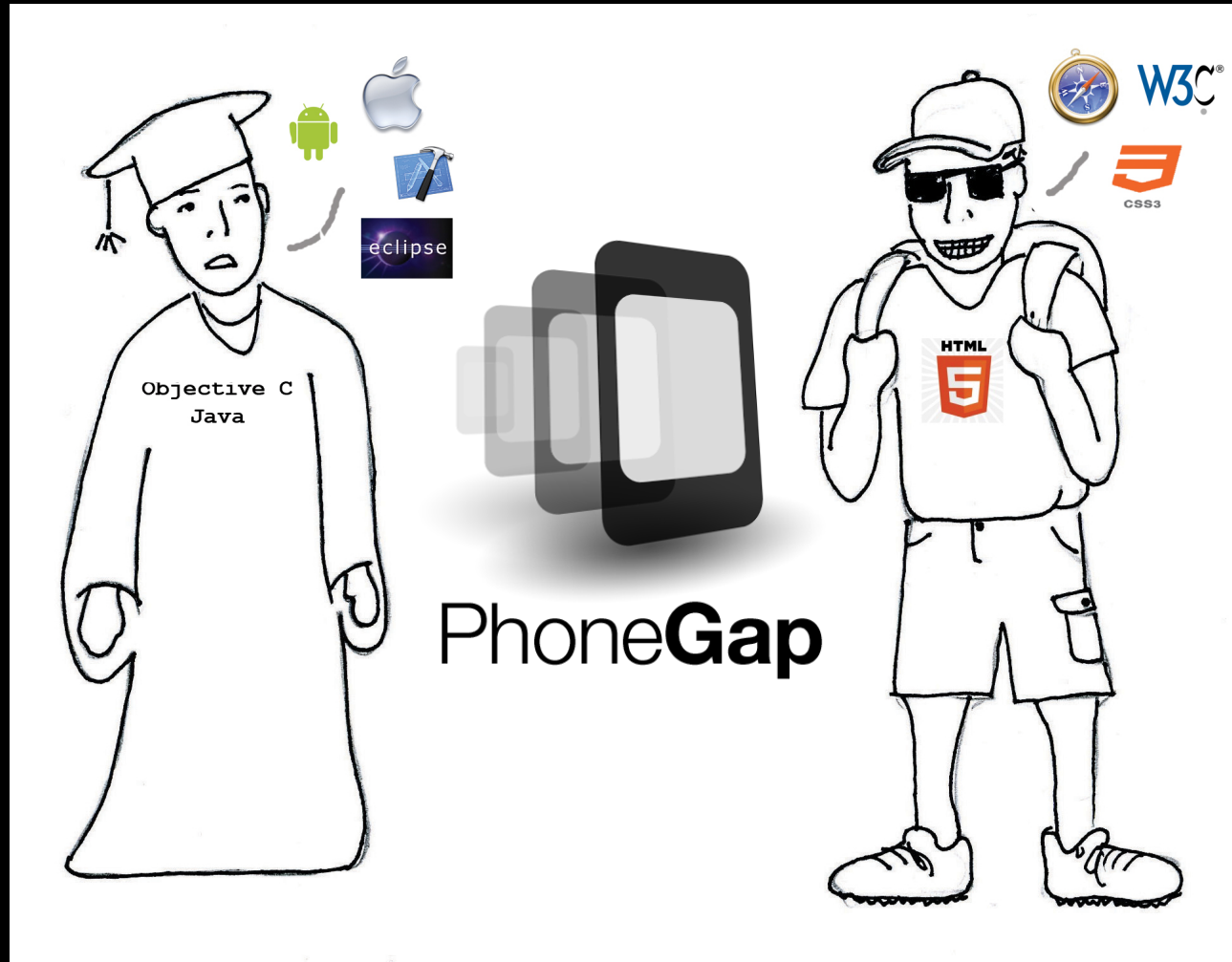- Licensing agreements

Hybrid Applications

- Built with Web Technologies
- Access to Device Features
- Distributed via App Stores
- Some platform specific knowledge
- Licensing Agreements

Mobile App Comparison

	Web	Hybrid	Native
Dev cost	Reasonable	Reasonable	Expensive
Dev time	Short	Short	Long
App portability	High	High	None
Performance	Fast	Native speed if needed	Very Fast
Native functionality	No	All	All
App store distribution	No	Yes	Yes
Extensible	No	Yes	Yes

What's a Developer to Do?

What is PhoneGap?

It's a tool for building mobile apps using Web technologies

HTML For Layout
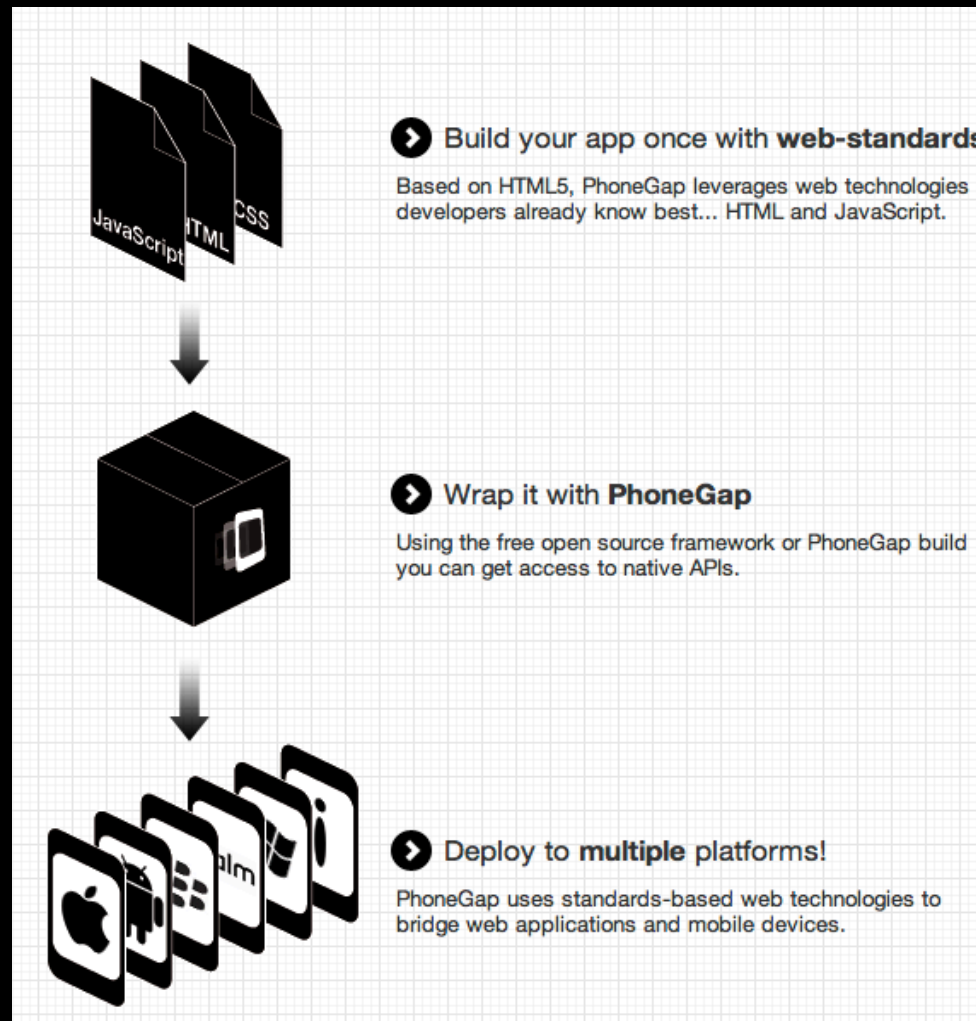JavaScript to access device functionality
navigator.notification.vibrate()
CSS for rich look and feel
Standards Based

IBM

How Does PhoneGap Work?

> **Build your app once with web-standards**

Based on HTML5, PhoneGap leverages web technologies developers already know best... HTML and JavaScript.

> **Wrap it with PhoneGap**

Using the free open source framework or PhoneGap build you can get access to native APIs.

> **Deploy to multiple platforms!**

PhoneGap uses standards-based web technologies to bridge web applications and mobile devices.

PhoneGap Community

- **Started by Nitobi in 2008**
- **Large, active open-source community**
- **350,000 downloads, 175,000 visits/month, 5000+ forum members**
- **Open Source – http://github.com/phonegap**
  - Apache2 inbound (CLA)
  - MIT & BSD outbound
- **Companies participating: IBM, RIM, Microsoft**
- **1.0 Release July, 2011**

PhoneGap Apps

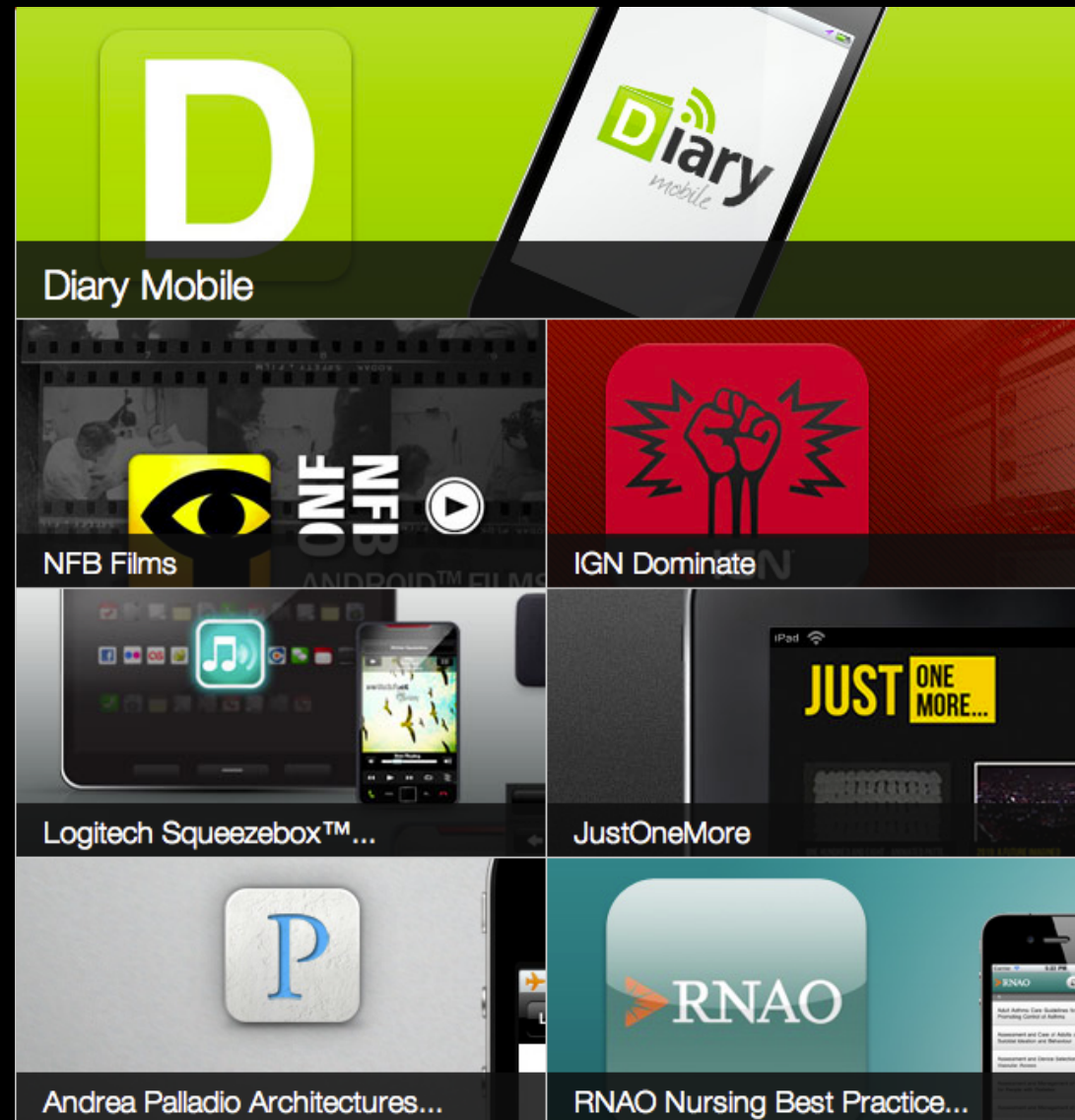## How many are there?

You can't tell!
(unless you are told)

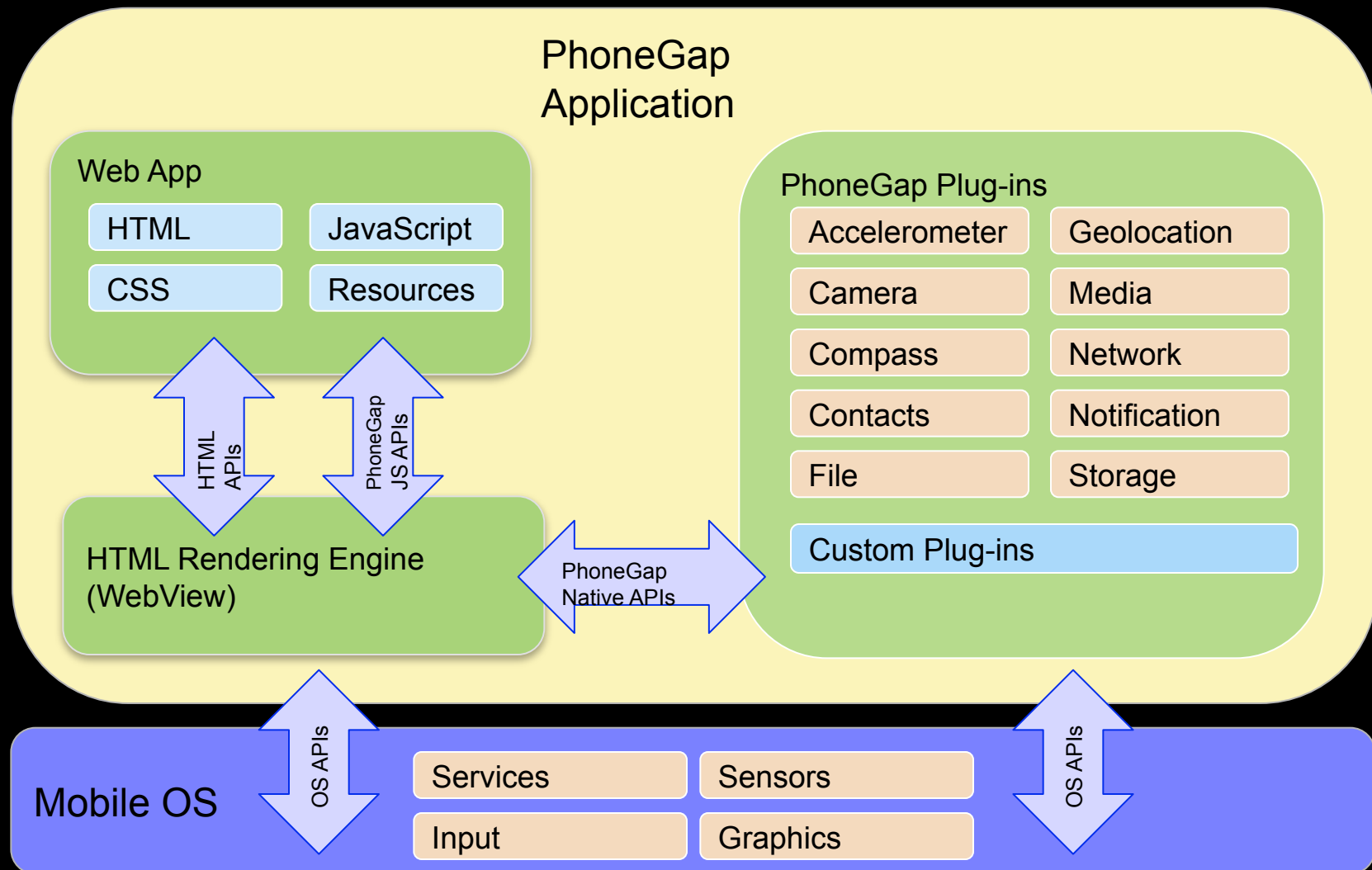http://www.phonegap.com/apps

IBM

PhoneGap Features

Core APIs

- Media (audio play/record)
- Media Capture: audio, video, camera – (W3C)
- Network (W3C)
- Notification
- Storage (HTML5 Web SQL & Web Storage)

- Accelerometer
- Compass
- Camera
- Contacts (W3C)
- File (W3C)
- File Transfer
- Geolocation (HTML5)

Typical Architecture

- HTML/JS/CSS + graphic assets are on the device, packaged as part of the build process.
- JavaScript can store retrieved data in a SQLite database or from localStorage (key/value pair) for offline access.
- Often a server component involved.
- JavaScript communicates with the server via XHR to get retrieve data.
- Yes, you can load your app remotely with caveats.

PhoneGap Architecture Diagram

PhoneGap Application

Web App

HTML	JavaScript
CSS	Resources

HTML APIs

PhoneGap JS APIs

HTML Rendering Engine (WebView)

PhoneGap Native APIs

PhoneGap Plug-ins

Accelerometer	Geolocation
Camera	Media
Compass	Network
Contacts	Notification
File	Storage

Custom Plug-ins

OS APIs

Mobile OS

Services	Sensors
Input	Graphics

OS APIs

Writing for Multiple Devices

Use a JavaScript Toolkit!
Most support at least Android and iOS
Remember that size matters!

Sencha Touch

Mobile Dojo

Best Practices

- Where possible use single HTML page
  - Use JavaScript to show/hide page elements based on user interaction instead of linking to a separate page.
  - Those JavaScript toolkits come in handy here!
- Obfuscate / crunch your JavaScript before release
- Consider offline usage
  - navigator.network.connection.type
  - online /offline events
  - Note: Apple checks for offline support
- Utilize persistent storage
  - File APIs
  - Storage APIs
  - SQL Lite / Lawnchair

Using PhoneGap

1. Create resources in www directory
2. Include phonegap.*.js in your start page (usually index.html)
3. Create onload handler
4. In onload handler register for deviceReady event from PhoneGap
5. When deviceready fires PhoneGap is ready.
6. Build, install and test for each platform

Building a PhoneGap App

DEMO

Debugging?

Console.log is your friend!

Wednesday @ 4:10pm – Portland 252
Debugging Mobile Web Applications with weinre
Patrick Mueller (IBM)

Extending PhoneGap

- Provide native code and JavaScript API
  - Extend Existing Features
  - Support for new sensors (NFC, ...)
- Examples of current plug-ins
  - http://github.com/phonegap/phonegap-plugins
  - Barcode scanner (Android, iOS)
  - Clipboard manager (Android, iOS)
  - Child browser (Android, iOS)
  - Contact viewer (Android)
  - iAd (iOS)
  - PayPal (iOS)

Future

- Additional Features
    - Plugin Only Architecture
    - W3C APIs
        - Web Notifications
        - Calendar
        - Messaging
    - Crypto / security
    - Menus
    - More at
      http://wiki.phonegap.com/w/page/28291160/roadmap-planning
- We want YOU to contribute code!
    - Bug fixes
    - Plugins

Other PhoneGap folks presenting @ OSCON

Wednesday @ 5:00pm - Portland 252
**Android WebKit Development - A
Cautionary Tale**
Joe Bowser (Nitobi)

Thursday @2:30pm – Portland Ballroom
**HTML5 and Mobile: Getting from Good
Enough to Great**
Filip Maj (Nitobi)

Resources

- **More Info**
  - http://www.phonegap.com/

- **Getting Started**
  - http://www.phonegap.com/start

- **Mailing List / Google Group**
  - http://groups.google.com/group/phonegap

- **API Documentation**
  - http://docs.phonegap.com/

- **Wiki**
  - http://wiki.phonegap.com/w/page/16494772/FrontPage

- **CODE**
  - http://github.com/phonegap