Web-based Hybrid Mobile Apps: State of the Practice and Research Opportunities

Austin, 17th May 2016

Ivano Malavolta
Assistant professor, Vrije Universiteit Amsterdam
i.malavolta@vu.nl
Roadmap

Questions

Who is speaking?

Web-based hybrid mobile apps

Hybrid apps in the Google Play store

Interactive session with Apache Cordova

Research opportunities and discussion
About me

- **2008**: Started PhD
- **2009**: iOS developer
- **2010**: Android developer
- **2011**: Cross-platform developer
- **2012**: Course started
- **2014**: Started doing research on mobile apps
- **TODAY**: Research on mobile-enabled systems (performance + security + sustainability)

Model-driven Engineering applied to software architecture

Had to design a course on mobile apps development
Roadmap

Who is speaking?

Web-based hybrid mobile apps

Hybrid apps in the Google Play store

Interactive session with Apache Cordova

Research opportunities and discussion
Developing a mobile strategy

Mobile strategy =

How much it will cost you to develop your app?

How much time?

How much effort?

How much money?
Bursting the first myth

Mobile design and development **IS NOT CHEAP**

$32,639 for a simple app
- 2 months to create
- 2 weeks per feature
- 4 features

$8,160 per feature

$163,200 for a complex app
- 6 months to create
- 2 weeks per feature
- 12 features

$13,600 per feature

[http://www.slideshare.net/fling/native-v-hybrid-v-web](http://www.slideshare.net/fling/native-v-hybrid-v-web)
Native VS web VS hybrid

Native

Web

Hybrid

Native App

BROWSER

NATIVE WRAPPER

PLATFORM APIs

PLATFORM APIs

<html>
<head>
<script src="...">
</head>
<body>
...
</body>
</html>
Native

**PROS**

- Lets you create apps with rich user interfaces and/or heavy graphics

**CONS**

- Development time
- Development cost
- Ongoing maintenance
- No portability (apps cannot be deployed on other platforms)

Main issue

FRAGMENTATION ➔ a native mobile app is written from scratch for each platform

Objective -C code
Swift code
XCode

Java code
C++ code
Eclipse

C# code
C++ code
JS code
Visual Studio

Native app
service requests
iOS

Native app
service requests
Android

Native app
service requests
Windows Phone
Examples of native apps

http://www.whatsapp.com/
http://www.ea.com/it/ipad/nfs
Web

**PROS**

- Offers fast development, simple maintenance, and full application portability
- One mobile web app works on any platform

**CONS**

- Can’t handle heavy graphics
- Can’t access low level features
  - (e.g., camera)

http://bit.ly/GWOaP1
Examples of mobile web apps

http://amazon.com
http://asidemag.com
Hybrid

**PROS**

- cross-platform *portability*
- *reuse* of existing knowledge of web developers
- *simpler* and less expensive development processes

**CONS**

- restricted access to *hardware* features
- decrease in *performance*
- variations on *user experience*
Web-based hybrid mobile apps

Recurrent architecture:
– apps are developed using **standard web technologies**
– on top of a hybrid development framework
  • providing a **native wrapper** and a generic JavaScript API that **bridges** all the service requests to the corresponding platform API
Examples of hybrid apps

http://sworkit.com
## Comparison

<table>
<thead>
<tr>
<th></th>
<th>SKILLS/TOOLS</th>
<th>DISTRIBUTION</th>
<th># APPS TO REACH MAJOR PLATFORMS</th>
<th>MAINTENANCE</th>
<th>DEVICE ACCESS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Native</strong></td>
<td>Objective-C, Java, C, C++, C#, VB.net</td>
<td>App Store/Market</td>
<td>4</td>
<td>Difficult</td>
<td>Full (Camera, microphone, GPS, gyroscope, file upload, accelerometer, etc.)</td>
</tr>
<tr>
<td><strong>Hybrid</strong></td>
<td>HTML, CSS, Javascript + mobile app development framework</td>
<td>App Store/Market</td>
<td>1</td>
<td>Moderate</td>
<td>Full (Camera, microphone, GPS, gyroscope, file upload, accelerometer, etc.)</td>
</tr>
<tr>
<td><strong>Web</strong></td>
<td>HTML, CSS, Javascript</td>
<td>Internet</td>
<td>1</td>
<td>Low</td>
<td>Partial (GPS, gyroscope, accelerometer, file upload)</td>
</tr>
</tbody>
</table>
Another perspective

Native
- Advanced UI interactions
- Fastest performance
- App store distribution

Hybrid
- Web developer skills
- Access to native platform
- App store distribution

single platform
full capability
multiple platforms
partial capability

HTML
- Web developer skills
- Instant updates
- Unrestricted distribution

http://goo.gl/KOdxh
My vision

Mobile web seems to be the only long-term commercially viable content platform for mobile devices

- FRAGMENTATION
- THE WEB
- USER EXPECTATIONS
- WEB UBIQUITY
My vision

FRAGMENTATION

When you go native there are too many platforms to be supported

- Symbian
- Android
- Maemo.org
- BlackBerry
- Windows Phone
- Palm webOS
My vision

THE WEB

The only medium for information and services that lasts from over 15 years

The web is an advanced technology
- webGL
- Local storage management
- Positioning & mapping
- Real-time data
- Push, ...
My vision

USER EXPECTATIONS

Users expect things to just work

- they don’t care about what platform they have
- they simply expect that your app will be available for their device

→ YOU HAVE TO BE CROSS-PLATFORM
The web is the only platform that

- works *across devices*
- apps share the same set of *standards*
- the same app can work also on a *desktop*
Wrap up

Mobile app technologies are **constantly moving targets**...
- new OSs, new versions of OSs, new UI patterns, new frameworks, etc.

Choosing between native vs web vs hybrid depends on a lot of factors, such as
- Which type of app are you creating?
- When do you need it?
- What are your skills?
Roadmap

Who is speaking?

Web-based hybrid mobile apps

Hybrid apps in the Google Play store

Interactive session with Apache Cordova

Research opportunities and discussion
Two empirical studies

**STUDY 1**
Developer creates App

**STUDY 2**
App download & use End users

**S1**
Are hybrid mobile apps published in the Google Play Store? What are their main traits?

**S2**
What is the difference between hybrid and native mobile apps as perceived by end users?
Design of the study

We analysed hybrid mobile apps
• in their actual context of use
• with a reproducible empirical strategy
  – well-defined empirical protocol
  – dataset comprising 11,917 real apps
    and 3,041,315 user reviews
  – dedicated analysis process and tool

Complete replication packages:
• http://cs.gssi.infn.it/ms_2015
• http://cs.gssi.infn.it/hybrid googleplay analysis
Analysis tool: http://github.com/GabMar/ApkCategoryChecker
Data extraction

Google Play

Apps and reviews mining

~11k app binaries

Hybrid apps classifier

~3M user reviews

Reviews analyzer

50 pages (~255) of reviews for each app

Top-500 most popular free apps for each category of the Google Play Store

Classified apps (hybrid vs native)

App scores

- Perceived value: 0.5
- Users sentiment: 0.6
- #reviews: 243
- Performance: 0.6
- Bugginess: 0.1
- Size: 3,456 kb
Where are hybrid mobile apps?

<table>
<thead>
<tr>
<th>Category</th>
<th>Total</th>
<th>Native</th>
<th>Hybrid (%)</th>
<th>Apache Cordova</th>
<th>Appcelerator</th>
<th>PhoneGap</th>
<th>Sencha</th>
<th>Kivy</th>
<th>Rho Mobile</th>
<th>IUI</th>
<th>Enyo</th>
</tr>
</thead>
<tbody>
<tr>
<td>Finance</td>
<td>463</td>
<td>410</td>
<td>53 (11.45)</td>
<td>23</td>
<td>29</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Medical</td>
<td>490</td>
<td>445</td>
<td>45 (9.18)</td>
<td>24</td>
<td>11</td>
<td>2</td>
<td>6</td>
<td>2</td>
<td></td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>Transportation</td>
<td>438</td>
<td>404</td>
<td>34 (7.76)</td>
<td>18</td>
<td>11</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Travel &amp; Local</td>
<td>484</td>
<td>450</td>
<td>34 (7.02)</td>
<td>23</td>
<td>5</td>
<td>4</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Health &amp; Fitness</td>
<td>352</td>
<td>331</td>
<td>21 (5.97)</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Libraries &amp; Demo</td>
<td>418</td>
<td>410</td>
<td>8 (5.97)</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Business</td>
<td>488</td>
<td>459</td>
<td>29 (5.94)</td>
<td>17</td>
<td>3</td>
<td>3</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lifestyle</td>
<td>497</td>
<td>468</td>
<td>29 (5.84)</td>
<td>12</td>
<td>11</td>
<td>5</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Social</td>
<td>491</td>
<td>465</td>
<td>26 (5.30)</td>
<td>16</td>
<td>3</td>
<td>5</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sports</td>
<td>497</td>
<td>473</td>
<td>24 (4.83)</td>
<td>15</td>
<td>6</td>
<td>2</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Shopping</td>
<td>487</td>
<td>464</td>
<td>23 (4.72)</td>
<td>12</td>
<td>6</td>
<td>3</td>
<td>1</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Education</td>
<td>493</td>
<td>473</td>
<td>20 (4.06)</td>
<td>14</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Book &amp; References</td>
<td>472</td>
<td>457</td>
<td>15 (3.18)</td>
<td>12</td>
<td>2</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Communication</td>
<td>487</td>
<td>471</td>
<td>16 (3.29)</td>
<td>13</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Entertainment</td>
<td>487</td>
<td>472</td>
<td>15 (3.08)</td>
<td>10</td>
<td>2</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>News &amp; Magazines</td>
<td>491</td>
<td>478</td>
<td>13 (2.65)</td>
<td>2</td>
<td>11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Comics</td>
<td>465</td>
<td>456</td>
<td>9 (1.94)</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Weather</td>
<td>495</td>
<td>488</td>
<td>7 (1.41)</td>
<td>4</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Media &amp; Video</td>
<td>483</td>
<td>477</td>
<td>6 (1.24)</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Productivity</td>
<td>495</td>
<td>489</td>
<td>6 (1.21)</td>
<td>3</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>Photography</td>
<td>494</td>
<td>489</td>
<td>5 (1.01)</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Music &amp; Audio</td>
<td>477</td>
<td>473</td>
<td>4 (0.84)</td>
<td>1</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tools</td>
<td>489</td>
<td>486</td>
<td>3 (0.61)</td>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Game</td>
<td>492</td>
<td>491</td>
<td>1 (0.20)</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Personalization</td>
<td>493</td>
<td>492</td>
<td>1 (0.20)</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>ALL</strong></td>
<td><strong>11,917</strong></td>
<td><strong>11,470</strong></td>
<td><strong>445 (3.73)</strong></td>
<td><strong>258</strong></td>
<td><strong>116</strong></td>
<td><strong>37</strong></td>
<td><strong>23</strong></td>
<td><strong>4</strong></td>
<td><strong>3</strong></td>
<td><strong>3</strong></td>
<td><strong>1</strong></td>
</tr>
</tbody>
</table>

Data-intensive mobile apps[2]

Apps with closer interaction with the Android platform

Winners, in line with informal claims
### Results – study 1

#### Findings - RQ1

**Presence in the Google Play Store**

**Finding**

3.73% hybrid mobile apps (445 over 11917)

- significantly *uncommon among the top-500* apps within 25 Google Play categories
- hybrid apps are infrequent in categories requiring *closer interaction with the Android platform and hardware*: Photography, Music, Audio, Tools, Game and Personalization

**Note**

Clear indicator of a future area of work for developers and vendors of hybrid development frameworks
Results – study 1

Findings - RQ2

Used hybrid development frameworks

Finding

Clear winners: Apache Cordova (258) and Appcelerator Titanium (116)

- other frameworks are less used
- in line with informal claims in other research papers [MCM^+13, OT12, CSS12]

Note

Appcelerator Titanium has a spike in the Finance category. It has specific support for enterprise-level authentication and authorization
Results – study 1

Findings - RQ3

<table>
<thead>
<tr>
<th>#</th>
<th>Web library</th>
<th># apps</th>
<th>#</th>
<th>Web library</th>
<th># apps</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>jQuery</td>
<td>267</td>
<td>11</td>
<td>Underscore</td>
<td>29</td>
</tr>
<tr>
<td>2</td>
<td>jQuery Mobile</td>
<td>106</td>
<td>12</td>
<td>Backbone</td>
<td>27</td>
</tr>
<tr>
<td>3</td>
<td>Json2</td>
<td>99</td>
<td>13</td>
<td>Jasmine</td>
<td>27</td>
</tr>
<tr>
<td>4</td>
<td>Ionic</td>
<td>58</td>
<td>14</td>
<td>Lo-Dash</td>
<td>21</td>
</tr>
<tr>
<td>5</td>
<td>AngularJS</td>
<td>55</td>
<td>15</td>
<td>RequireJS</td>
<td>21</td>
</tr>
<tr>
<td>6</td>
<td>Google Analytics</td>
<td>38</td>
<td>16</td>
<td>Bootstrap</td>
<td>20</td>
</tr>
<tr>
<td>7</td>
<td>Fastclick</td>
<td>35</td>
<td>17</td>
<td>Mobiscroll</td>
<td>20</td>
</tr>
<tr>
<td>8</td>
<td>jQuery UI</td>
<td>32</td>
<td>18</td>
<td>Crypto-js</td>
<td>16</td>
</tr>
<tr>
<td>9</td>
<td>Moment.js</td>
<td>32</td>
<td>19</td>
<td>Datejs</td>
<td>15</td>
</tr>
<tr>
<td>10</td>
<td>Facebook SDK</td>
<td>30</td>
<td>20</td>
<td>TweenJS</td>
<td>14</td>
</tr>
</tbody>
</table>
Results – study 1

Integration to the Android platform and other apps

Finding

*hybrid* mobile apps are generally *in line with native* mobile apps

- 17 out of the top 20 permissions are requested to the *Android* platform, whereas the remaining 3 permissions are requested to *Google services*
  - access the *Internet* and the *network* connection state
  - write to an *external storage* drive (e.g. an SD card)
  - *geographic position* of the user
Results – study 2

Results – value (RQ1)

Average of the ratings as provided by end users

Rating - real number in [1, 5]

Certain balance, with neglectable differences
Results – study 2

Results – value (RQ1)

Polarity of sentiment of end users

Non data-intensive or requiring multimedia capabilities

Balance between hybrid and native apps, with some exceptions

\[ \text{Polarity}_a = \frac{\text{pos}_a - \text{neg}_a}{\text{pos}_a + \text{neg}_a} \]

where $\text{pos}_a =$ #reviews with positive sentiment
$\text{neg}_a =$ #reviews with negative sentiment
Results – study 2

Results – performance (RQ2)

\[ \text{Performance}_{a} = \frac{\text{pos}_{a} - \text{neg}_{a}}{\text{pos}_{a} + \text{neg}_{a}} \]

where \( \text{pos}_{a} = \) #reviews with positive sentiment w.r.t. performance of the app
\( \text{neg}_{a} = \) #reviews with negative sentiment w.r.t. performance of the app

Balance between hybrid and native apps, with some exceptions
Results – study 2

The highest unbalance between the two development strategies in our study

Possible interpretation: absence of full-fledged testing frameworks for hybrid apps, such as those provided by native apps IDEs like Eclipse and Android Studio.

\[ \text{bugginess}_a = \frac{\text{bug}_a}{\text{reviews}_a} \]

where \( \text{bug}_a \) = \# reviews signalling the presence of bugs or failures
\( \text{reviews}_a \) = total number of reviews of the app
Results – initial download size (RQ4)

In line with the average size of Android apps [7]

\[ \text{size}_a = \text{file size in kilobytes of the app APK file} \]
Wrap up

- **3.73%** hybrid apps (445 over 11917)
  - Top: Apache Cordova (258) and Appcelerator Titanium (116)
- Developers **reuse JavaScript frameworks** already existing in the desktop web arena
  - also MVC frameworks like Angular and Backbone
- Hybrid apps request the **same permissions** of native apps
- End users **value** hybrid and native apps similarly
- Hybrid may be good for **data-intensive apps**, whereas it performs poorly when dealing with low-level, platform-specific features
- In some categories, native apps are perceived as better with respect to **performance and bugginess**
References


Ivano Malavolta, Stefano Ruberto, Valerio Terragni, Tommaso Soru, "End Users' Perception of Hybrid Mobile Apps in the Google Play Store". Mobile Services (MS), 2015 IEEE International Conference on. IEEE, 2015.
Roadmap

Who is speaking?

Web-based hybrid mobile apps

Hybrid apps in the Google Play store

Interactive session with Apache Cordova

Research opportunities and discussion
PhoneGap VS Cordova

PhoneGap is a distribution of Apache Cordova

Adobe/Nitobi donated the PhoneGap codebase to the Apache foundation

➔ wider audience and contributors
➔ transparent governance
Better documentation
➔ easier contributions for companies
Apache Licensing

There was only one problem....
trademark ambiguity ➔ CORDOVA
Intuition

The UI layer is a web browser
- 100% width
- 100% height

Headless web browser
- No URL bar
- No decorations
- No zooming
- No text selection
How does Cordova work?

- Cordova Javascript
- Mobile app
- API provider for Android
- API provider for iOS
- API provider for Windows Phone
- API provider for Blackberry
Cordova architecture

Cordova Application

Web App
- HTML
- config.xml
- JS
- CSS
- Resources

Cordova Plugins
- Accelerometer
- Geolocation
- Camera
- Media
- Device
- Network
- Contacts
- Storage

HTML Rendering Engine (WebView)

Custom Plugins

OS APIs
- Mobile OS
  - Services
  - Input
  - Sensors
  - Graphics
## Cordova APIs

<table>
<thead>
<tr>
<th></th>
<th>amazon-fires OS</th>
<th>android</th>
<th>blackberry10</th>
<th>Firefox OS</th>
<th>ios</th>
<th>Ubuntu</th>
<th>windows 8.0, 8.1, 10, Phone 8.1</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>cordova CLI</strong></td>
<td>🎁 Mac, Windows, Linux</td>
<td>🎁 Mac, Windows, Linux</td>
<td>🎁 Mac, Windows, Linux</td>
<td>🎁 Mac, Windows, Linux</td>
<td>🎁 Mac</td>
<td>🎁 Ubuntu</td>
<td>🎁 Windows</td>
</tr>
<tr>
<td><strong>Embedded WebView</strong></td>
<td>🎁 (see details)</td>
<td>🎁 (see details)</td>
<td>✗</td>
<td>✗</td>
<td>✗ (see details)</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td><strong>Plug-in Interface</strong></td>
<td>🎁 (see details)</td>
<td>🎁 (see details)</td>
<td>🎁 (see details)</td>
<td>✗</td>
<td>🎁 (see details)</td>
<td>🎁 (see details)</td>
<td>🎁</td>
</tr>
</tbody>
</table>

### Platform APIs

<table>
<thead>
<tr>
<th>Feature</th>
<th>Cordova CLI</th>
<th>android</th>
<th>blackberry10</th>
<th>Firefox OS</th>
<th>ios</th>
<th>Ubuntu</th>
<th>windows 8.0, 8.1, 10, Phone 8.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accelerometer</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
</tr>
<tr>
<td>BatteryStatus</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>✗</td>
<td>🎁 Windows Phone 8.1 only</td>
</tr>
<tr>
<td>Camera</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
</tr>
<tr>
<td>Capture</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>✗</td>
<td>🎁</td>
<td>🎁</td>
</tr>
<tr>
<td>Compass</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>✗</td>
<td>🎁</td>
<td>🎁 (3GS+)</td>
</tr>
<tr>
<td>Connection</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>✗</td>
<td>🎁</td>
<td>🎁</td>
</tr>
<tr>
<td>Contacts</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>✗</td>
<td>🎁</td>
<td>🎁 partially</td>
</tr>
<tr>
<td>Device</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
</tr>
<tr>
<td>Events</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>✗</td>
<td>🎁</td>
<td>🎁</td>
</tr>
<tr>
<td>File</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>✗</td>
<td>🎁</td>
<td>🎁</td>
</tr>
<tr>
<td>File Transfer</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>✗</td>
<td>🎁</td>
<td>🎁</td>
</tr>
<tr>
<td>Geolocation</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
</tr>
<tr>
<td>Globalization</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>✗</td>
<td>🎁</td>
<td>🎁</td>
</tr>
<tr>
<td>InAppBrowser</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>✗</td>
<td>🎁</td>
<td>🎁 uses iframe</td>
</tr>
<tr>
<td>Media</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>✗</td>
<td>🎁</td>
<td>🎁</td>
</tr>
<tr>
<td>Network</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>✗</td>
<td>🎁</td>
<td>🎁</td>
</tr>
<tr>
<td>SplashScreen</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>✗</td>
<td>🎁</td>
<td>🎁</td>
</tr>
<tr>
<td>Status Bar</td>
<td>✗</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>✗</td>
<td>🎁</td>
<td>🎁 Windows Phone 8.1 only</td>
</tr>
<tr>
<td>Storage</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>✗</td>
<td>🎁</td>
<td>🎁 (localStorage &amp; indexedDB)</td>
</tr>
<tr>
<td>Vibration</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>🎁</td>
<td>✗</td>
<td>🎁</td>
<td>🎁 Windows Phone 8.1 only</td>
</tr>
</tbody>
</table>
When Cordova APIs are not enough...

Sometimes Cordova is not enough as is for our purposes:

- unsupported feature
- heavyweight data processing is faster in native code
  - ex. images manipulation
- background processing is better handled natively
  - ex. files sync

⇒ You can develop a Cordova plugin
The Plugin Registry

plugreg allows Cordova / PhoneGap developers to search for existing plugins for their app projects. It also gives plugin authors additional exposure to their open source plugin(s).

At a glance, users are able to see the status of a plugin, the number of stars, open issues, version, supported platforms / engines and more.

There are currently 1591 plugins from 1058 different authors and growing all the time. Please submit your own plugin(s) if you haven’t already.
Jboss Hybrid Mobile Tools

Overview
Name and Description
ID: com.example.myapp
Name: My Application
Version: 2.0.0
Description: A sample Apache Cordova application that responds to the deviceready event.
Content Source: index.html

Author
Name: JBoss by Red Hat
Email: aerogear-dev@lists.jboss.org
Example of Cordova API usage (1)

```javascript
var options = {
    enableHighAccuracy: true,
    maximumAge: 3000,
    timeout: 5000
};

navigator.geolocation.getCurrentPosition(win, fail, options);

function win(pos) {
    var el = '<div>Latitude: ' + pos.coords.latitude + '</div>;
    el += '<div>Longitude: ' + pos.coords.longitude + '</div>;
    el += '<div>timestamp: ' + pos.timestamp + '</div>;
    $('#block').html(el);
}

function fail(err) {
    console.log(err.code);
}
```

http://i0.wp.com/www.phonegap.co.in/wp-content/uploads/2015/07/device_axes.jpg
Example of Cordova API usage (2)

```javascript
var options = { frequency: 3000 };  
var watchID = navigator.accelerometer.watchAcceleration(win, fail, options);

function win(acc) {
  if((acc.x === 0) && (acc.y === 0) && (acc.z === 9.81)) {
    console.log('I am on a table');
    stop();
  } else {
    console.log('Please, leave me on the table');
  }
}

function fail() {
  alert('error');
}

function stop() {
  if(watchID) {
    navigator.accelerometer.clearWatch(watchID);
    watchID = null;
  }
}
```

http://i0.wp.com/www.phonegap.co.in/wp-content/uploads/2015/07/device_axes.jpg
Interactive session

1. Create a new Apache Cordova app
2. Install the Geolocation and Camera plugins
3. Call the Nestoria REST API for getting a list of apartment listings
   – http://www.nestoria.co.uk/help/api
4. Develop a simple view for showing the apartments
5. Develop a simple view for adding a listing

The source code of this session is available here:
http://github.com/iivanoo/mobileSoft2016_interactiveSession
Roadmap

Who is speaking?

Web-based hybrid mobile apps

Hybrid apps in the Google Play store

Interactive session with Apache Cordova

Research opportunities and discussion
Cross-platform studies

What are the traits of hybrid apps on different platforms?

How the actual users perceive hybrid apps on different platforms?

RESEARCH INSTRUMENT
Cross-platform studies considering at least the three most popular app stores

- Google Play
- Apple iTunes
- Windows Phone
Investigate on extra-functional properties

The success of any application, mobile or otherwise, depends on a lengthy list of non-functional qualities. Among those most relevant to mobile applications are performance (efficient use of device resources, responsiveness, scalability), reliability (robustness, connectivity, stability), quality (usability, installability), and security. Many of these issues have been addressed for web applications, and that knowledge provides an excellent starting point for studying mobile application requirements.
Performance issues?

Are there recurrent design and code antipatterns impacting the performance of hybrid mobile apps?

Do best practices for the desktop web apply here?

RESEARCH INSTRUMENT (cross-platform) large-scale studies on apps already published in the app stores.
Testing

Test the App for Each Platform Separately. Our interviews reveal that our participants treat each platform completely separately when it comes to testing. Currently, there is no coherent method for testing a given mobile app across different platforms; being able to handle the differences at the UI level is seen as a major challenge.

Testing Apps for Multiple Platforms. Regarding the testing challenges, follow-up studies could focus on generating test cases for mobile apps. A centralized automatic testing system that generates a (different) test case for each target platform could be a huge benefit. While platform-specific features can be customized, core features could share the same tests. Thus, further research should focus on streamlining application development and testing efforts regardless of the mobile platform.

Behavioural consistency VS HCI guidelines

constantly faced with two competing requirements:

- **Familiarity for platform users:** Each platform follows a set of specific HCI guidelines to provide a consistent look-and-feel across applications on the same device. This makes it easier for end users to navigate and interact with various applications.

- **Behavioural consistency across platforms:** On the other hand, developers would like their application to behave similarly across platforms, e.g., user interaction with a certain feature on Blackberry should be the same as on iPhone and Android.

> "Different platforms have different strengths and possibilities. It is foolish to try to make the apps exactly the same between platforms"; and: "It requires multi-platform considerations at the designing stage and clever decisions should be made where platform-specific design is necessary."

Conclusions

Native VS web VS hybrid

Two empirical studies

STUDY 1
- Developer
- App
- End users

STUDY 2
- S1: Are hybrid mobile apps published in the Google Play Store? What are their main traits?
- S2: What is the difference between hybrid and native mobile apps as perceived by end users?

How does Cordova work?

Cordova Javascript

Investigate on extra-functional properties

The success of any application, mobile or otherwise, depends on a lengthy list of non-functional qualities. Among these most relevant to mobile applications are performance (efficient use of device resources, responsiveness, scalability), reliability (robustness, connectivity, stability, quality (stability, installability), and security. Many of these issues have been addressed for web applications, and that knowledge provides an excellent starting point for studying mobile application requirements.
Contact

Ivano Malavolta | Assistant professor
Vrije Universiteit Amsterdam

iivanoo

i.malavolta@vu.nl

www.ivanomalavolta.com