



Développement mobile

« cross-platform »

Polytech'Nice

vendredi 17 octobre 2014

Au programme ...

1. A chaque projet sa techno
2. 1 code → Plusieurs plateformes
3. La puissance de jQuery sur mobile
4. Quelques librairies mobiles intéressantes





Avant de commencer ...

Faisons les présentations

Qui suis-je ?



- Ingénieur Polytech'Nice - Promotion 2006
- Fondateur & Gérant d'i2N depuis 2007
- Consultant auprès des entreprises pour les accompagner dans leur gestion des technologies de l'information (web, mise en place de solutions logicielles & matérielle)

i2N : Notre métier



Conception de sites Internet sur mesure
(institutionnel, e-commerce, mobile)



Suivi et développement (référencement,
stratégie social media)



Développement de logiciels sur mesure



Gestion de parcs informatiques &
Maintenance

i2N : Nos références



Mairie de Monaco, Sûreté Publique de Monaco, SBM, Education Nationale, Grimaldi Forum, Radio Vitamine, Radio Riviera ...
Tous secteurs : Immobilier, BTP, Institutionnels, Juridique, Services ...

www.i2n.mc

Et vous ?

Quelques questions pour vous ...

Quelles sont vos connaissances en Javascript ?

Avez-vous déjà développé nativement pour mobile ?

Avez-vous déjà développé un site adapté aux mobiles ?





Bien choisir sa techno

Application ou site mobile,
Que choisir ?

Se poser les bonnes questions

- Que voulons nous faire avec notre mobile ?
 - Consultation de pages Internet
 - Consultation de données disponibles hors ligne
 - Consultation de données remises à jour
 - Interaction avec une application ou un site web
 - Utiliser les fonctionnalités du mobile (boussole, géolocalisation, accéléromètre)
 - Jeux en 3D
 - Streaming audio et vidéo

Avoir les bonnes réponses

- Que voulons nous faire avec notre mobile ?
 - Consultation de pages Internet → Site mobile
 - Consultation de données disponibles hors ligne
 - Site mobile / application
 - Consultation de données remises à jour
 - Site mobile / application
 - Interaction avec une application ou un site web
 - Site mobile / application
 - Utiliser les fonctionnalités du mobile (boussole, géolocalisation, accéléromètre)
 - Application Phonegap / native
 - Tirer parti du mobile à 100% (jeux en 3D, streaming audio et vidéo)
 - Application native

Le cross platform c'est bien

- **Mais** on n'a pas accès à toutes les fonctionnalités du mobile
- **Mais** on n'a pas une gestion aussi fine des performances
- **Mais** on n'est pas aussi rétro compatible qu'on le souhaiterait (quoi que)



Par contre, on **gagne du temps** si le choix est fait à bon escient





Phonegap / Apache Cordova

Des applis mobiles en Javascript ?

C'est possible ...

Vous avez dit Phonegap ?

- Le principe général :
 - 1 code unique
 - Plusieurs plateformes cibles
- Les avantages:
 - Temps de développement réduit
 - Maintenabilité sur différentes plateformes
 - 1 seul langage pour différentes plateformes



Phonégap ou Cordova ?

- Phonégap est une distribution de Cordova
 - Cordova est le moteur qui fait tourner Phonégap
 - Phonégap peut proposer quelques plugins complémentaires< (notamment propriétaires)
- Cependant ...
 - Ils restent tous les 2 gratuits
 - Ils restent tous les 2 open source



Vous avez dit Phonegap ?

- Recherche à réduire l'écart entre les téléphones
- Concrètement :
 - Outil de création d'un projet type pour chaque OS
 - 1 répertoire pour le code
 - Code HTML + Javascript
 - Librairie Javascript à inclure
 - Fichier de préférences à éditer suivant les besoins
- En pratique, Phonegap crée une WebView qui exécute du code Web

Vous avez dit Phonegap ?

- Cependant :
 - Quelques petits inconvénients liés à une WebView
 - N'évite pas les tests sur les différentes plateformes !
 - Difficile à déboguer finement car ça reste du JS ...
- Mais ces inconvénients sont à mettre en perspective par rapport au gain de temps et aux avantages !

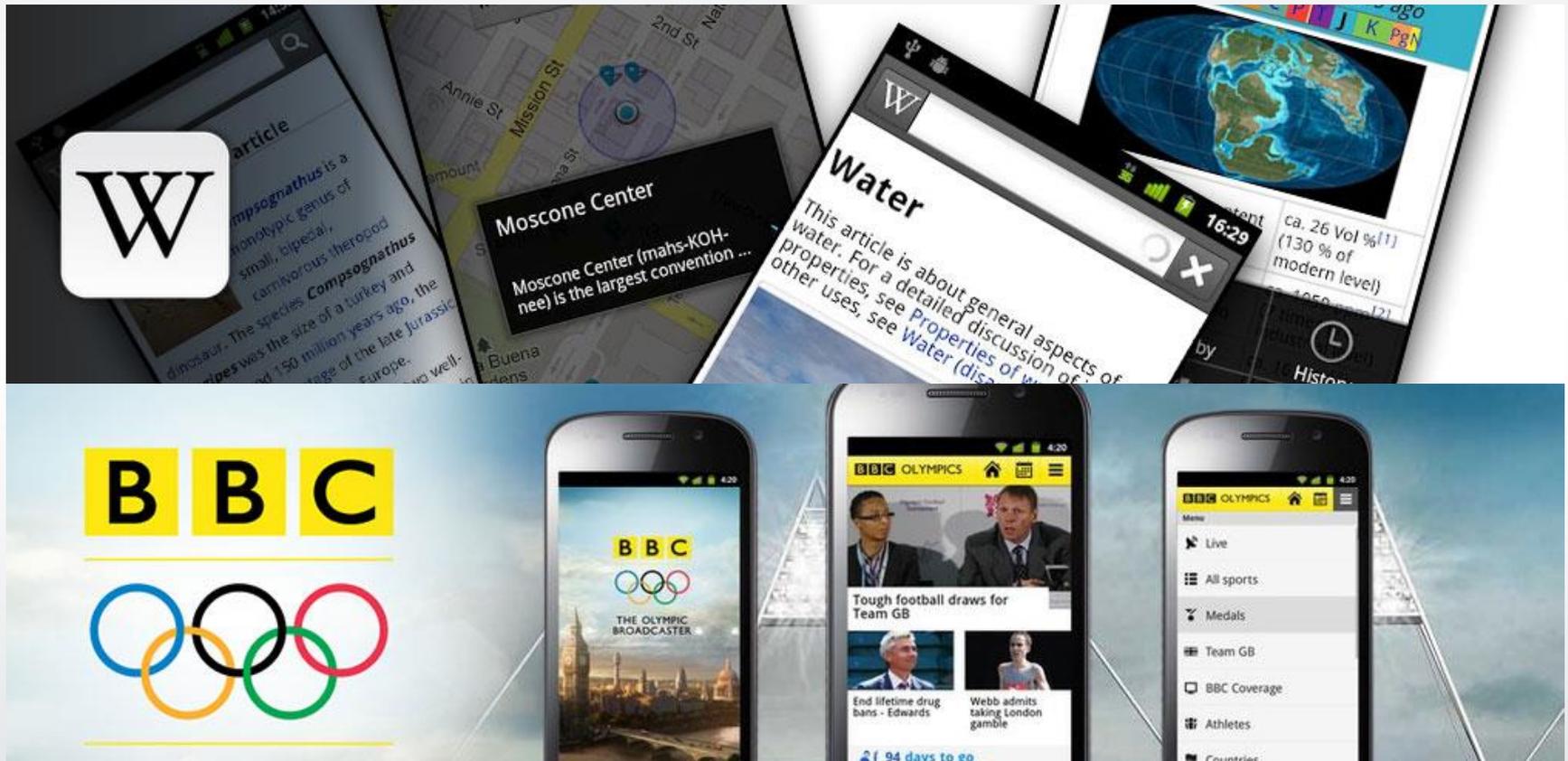
Ce qui est supporté

	iPhone / iPhone 3G	iPhone 3GS and newer	Android	Blackberry OS 6.0+	Blackberry 10	Windows Phone 8	Ubuntu	Firefox OS
Accelerometer	✓	✓	✓	✓	✓	✓	✓	✓
Camera	✓	✓	✓	✓	✓	✓	✓	✓
Compass	X	✓	✓	X	✓	✓	✓	✓
Contacts	✓	✓	✓	✓	✓	✓	✓	✓
File	✓	✓	✓	✓	✓	✓	✓	X
Geolocation	✓	✓	✓	✓	✓	✓	✓	✓
Media	✓	✓	✓	X	✓	✓	✓	X
Network	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Alert)	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Sound)	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Vibration)	✓	✓	✓	✓	✓	✓	✓	✓
Storage	✓	✓	✓	✓	✓	✓	✓	✓

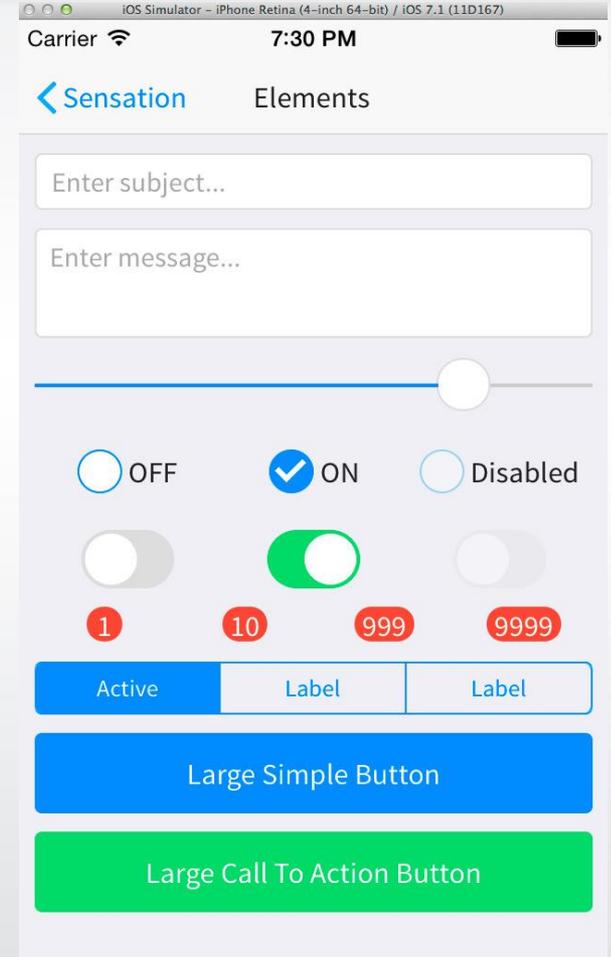
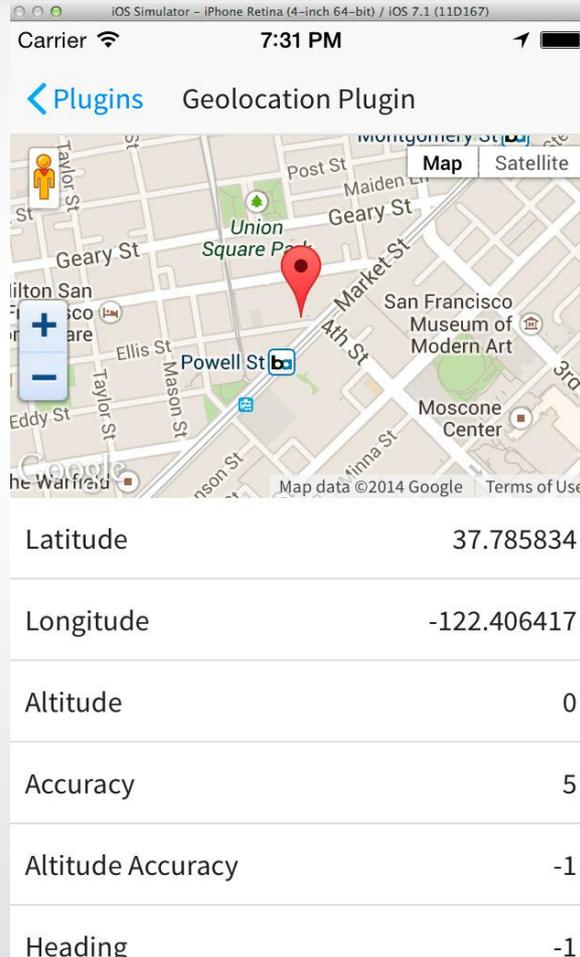
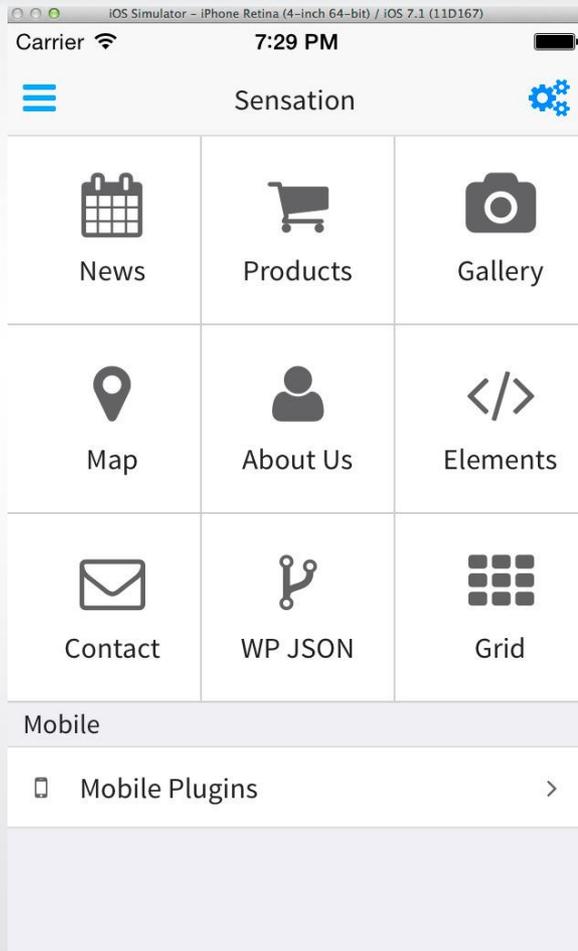
Ce qui est supporté

	android	blackberry (6)	blackberry10	ios	wp7 (Windows Phone 7)	wp8 (Windows Phone 8)	win8 (Windows 8)	firefoxos	tizen
cordova CLI	✓ Mac, Windows, Linux	✗ Mac, Windows	✓ Mac, Windows	✓ Mac	✓ Windows	✓ Windows	✗	✗	
Embedded WebView	✓ (see details)	✗	✗	✓ (see details)	✗	✗	✗	✗	
Plug-in Interface	✓ (see details)	✓ (see details)	✓ (see details)	✓ (see details)	✓ (see details)	✓	✗	✗	
Platform APIs									
Accelerometer	✓	✓	✓	✓	✓	✓	✓	✓	✓
Camera	✓	✓	✓	✓	✓	✓	✓	✓	✓
Capture	✓	✓	✓	✓	✓	✓	✗	✗	
Compass	✓	✗	✓	✓ (3GS+)	✓	✓	✓	✓	✓
Connection	✓	✓	✓	✓	✓	✓	✓	✓	✓
Contacts	✓	✓	✓	✓	✓	✓	✓	✗	✗
Device	✓	✓	✓	✓	✓	✓	✓	✓	✓
Events	✓	✓	✓	✓	✓	✓	✓	✓	✓
File	✓	✓	✓	✓	(partial)no FileTransfer	(partial)no FileTransfer	✓	✗	
Geolocation	✓	✓	✓	✓	✓	✓	✓	✓	✓
Globalization	✓	✓	✗	✓	✗	✓	✗	✗	✗
InAppBrowser	✓	✓	✓	✓	✓	✓	✗	✗	
Media	✓	✗	✓	✓	✓	✓	✓	✓	✓
Notification	✓	✓	✓	✓	✓	✓	✓	✓	✓
Splashscreen	✓	✗	✓	✓	✓	✓	✓	✗	
Storage	✓	✓	✓	✓	(partial) localStorage only	(partial) localStorage only	✓	✓	

Ce que ça donne



Ce que ça donne





Phonegap / Apache Cordova

Oui mais techniquement ça donne quoi ?

Installer Phoneygap

- Prérequis :
 - Node JS
 - SDK cible (Android, iOS ...)

```
$ sudo npm install -g phoneygap
```

```
$ phoneygap create my-app  
$ cd my-app  
$ phoneygap run android
```

Avec Cordova

```
$ sudo npm install -g cordova
```

```
$ cordova create hello com.example.hello HelloWorld
```

```
$ cd hello
```

```
$ cordova platform add ios  
$ cordova platform add android  
$ cordova platform add blackberry10  
$ cordova platform add firefoxos
```

```
$ cordova build ios
```

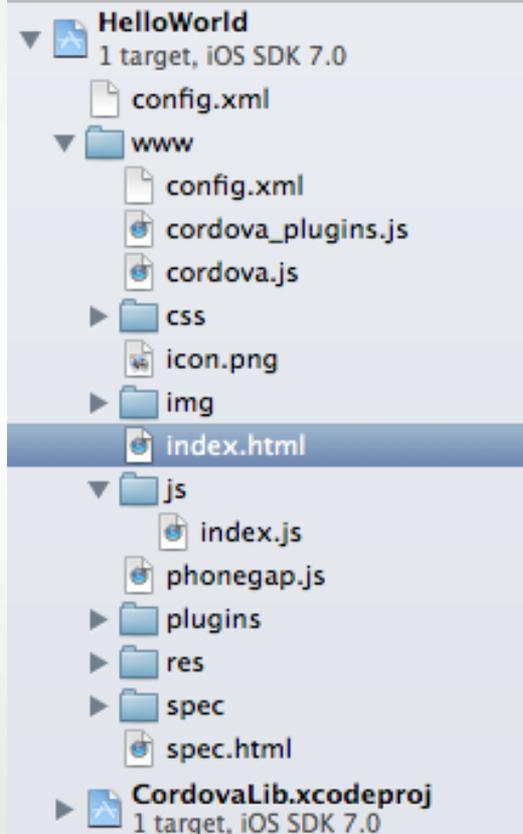
Exemple de projet

```
<!DOCTYPE HTML>
<html>
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, height=device-height, initial-scale=1, maximum-scale=1, user-scalable=no">
  <script src="components/loader.js"></script>
  <link rel="stylesheet" href="components/loader.css">
  <link rel="stylesheet" href="css/style.css">
  <script>
    // PhoneGap event handler
    document.addEventListener("deviceready", onDeviceReady, false);
    function onDeviceReady() {
      console.log("PhoneGap is ready");
    }
  </script>
</head>
<body>

  <h1>Welcome to Monaca!</h1>
  <a class="button--large" href="phonegap-demo.html">Start Demo</a>

</body>
</html>
```

Exemple de projet



```
<html>
  <head>
    <meta charset="utf-8" />
    <meta name="format-detection" content="telephone=no" />
    <meta name="viewport" content="user-scalable=no, initial-scale=1, maximum-
      scale=1, minimum-scale=1, width=device-width, height=device-height,
      target-densitydpi=device-dpi" />
    <link rel="stylesheet" type="text/css" href="css/index.css" />
    <title>Hello World</title>
  </head>
  <body>
    <div class="app">
      <h1>PhoneGap</h1>
      <div id="deviceready" class="blink">
        <p class="event listening">Connecting to Device</p>
        <p class="event received">Device is Ready</p>
      </div>
    </div>
    <script type="text/javascript" src="phonegap.js"></script>
    <script type="text/javascript" src="js/index.js"></script>
    <script type="text/javascript">
      app.initialize();
      //navigator.geolocation.getCurrentPosition(onSuccess, onError);
    </script>
  </body>
</html>
```

Le fichier config.xml

- Définition
 - des composants disponibles
 - du nom de l'application
 - du splashscreen
 - de paramètres globaux
 - ...
- Différent suivant les cibles

Le fichier config.xml (Android)

```
<?xml version="1.0" encoding="UTF-8"?>
<widget xmlns="http://www.w3.org/ns/widgets" id="com.example.helloworld" version="1.0.0">
  <name>Hello World App</name>
  <description>Hello World App</description>
  <author/>
  <access origin="*" />
  <preference name="loglevel" value="DEBUG" />
  <preference name="AndroidLaunchMode" value="singleTop" />
  <preference name="BackgroundColor" value="0xffffffff" />
  <preference name="DisallowOverscroll" value="true" />
  <preference name="ErrorUrl" value="" />
  <preference name="Fullscreen" value="false" />
  <preference name="KeepRunning" value="true" />
  <preference name="monaca:androidVersionCode" value="1" />
  <preference name="monaca:WebViewEngine" value="default" />
  <preference name="SplashScreen" value="screen" />
  <preference name="SplashScreenDelay" value="1000" />
  <preference name="Orientation" value="default" />
  <preference name="monaca:DisableCookie" value="false" />
  <preference name="monaca:AndroidIsPackageNameSeparate" value="false" />
  <feature name="App">
    <param name="android-package" value="org.apache.cordova.App" />
  </feature>
</widget>
```

Le fichier config.xml (iOS)

```
<?xml version="1.0" encoding="UTF-8"?>
<widget xmlns="http://www.w3.org/ns/widgets" id="com.example.helloworld" version="1.0.0">
  <name>HelloWorld</name>
  <description>Hello World App</description>
  <author/>
  <access origin="*" />
  <content src="index.html" />
  <preference name="AllowInlineMediaPlayback" value="false" />
  <preference name="AutoHideSplashScreen" value="true" />
  <preference name="BackupWebStorage" value="cloud" />
  <preference name="DisallowOverscroll" value="true" />
  <preference name="EnableViewportScale" value="false" />
  <preference name="FadeSplashScreen" value="true" />
  <preference name="FadeSplashScreenDuration" value=".25" />
  <preference name="KeyboardDisplayRequiresUserAction" value="true" />
  <preference name="MediaPlaybackRequiresUserAction" value="false" />
  <preference name="ShowSplashScreenSpinner" value="false" />
  <preference name="SuppressesIncrementalRendering" value="false" />
  <preference name="TopActivityIndicator" value="gray" />
  <preference name="GapBetweenPages" value="0" />
  <preference name="PageLength" value="0" />
  <preference name="PaginationBreakingMode" value="page" />
  <preference name="PaginationMode" value="unpaginated" />
  <feature name="LocalStorage">
    <param name="ios-package" value="CDVLocalStorage" />
  </feature>
  <preference name="monaca:targetFamilyiPhone" value="1" />
  <preference name="monaca:targetFamilyiPad" value="1" />
  <preference name="monaca:DisableCookie" value="false" />
  <preference name="monaca:iosApplicationName" value="Hello World App" />
</widget>
```

Le fichier config.xml

- <widget> : domaine de l'application
- <name> : nom de l'application
- <description> & <author> : métadonnées
- <content> : page de démarrage (défaut : index.html)
- <access> : domaines extérieurs auxquels l'application peut accéder

```
<widget id="com.example.hello" version="0.0.1">
  <name>HelloWorld</name>
  <description>
    A sample Apache Cordova application that responds to the deviceready event.
  </description>
  <author email="dev@callback.apache.org" href="http://phonegap.com">
    Apache Cordova Team
  </author>
  <content src="index.html" />
  <access origin="*" />
  <preference name="Fullscreen" value="true" />
  <preference name="WebViewBounce" value="true" />
</widget>
```

Quelques composants ...

- Accéléromètre
- Caméra
- Boussole
- Contacts
- Géolocalisation
- Lecteur multimédia
- Stockage
- Notifications
- ...

Utilisation de la géolocalisation

Avec l'outil en ligne de commande :

```
$ cordova plugin add org.apache.cordova.geolocation
$ cordova plugin ls
[ 'org.apache.cordova.geolocation' ]
$ cordova plugin rm org.apache.cordova.geolocation
```

Sous Android :

```
(in app/res/xml/config.xml)
<feature name="Geolocation">
  <param name="android-package" value="org.apache.cordova.GeoBroker" />
</feature>

(in app/AndroidManifest.xml)
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_LOCATION_EXTRA_COMMANDS" />
```

Utilisation de la géolocalisation

Sous iOS :

```
<feature name="Geolocation">  
  <param name="ios-package" value="CDVLocation" />  
</feature>
```

Sous Windows :

```
<Capabilities>  
  <Capability Name="ID_CAP_LOCATION" />  
</Capabilities>
```

Utilisation de la géolocalisation

```
// onSuccess Callback
// This method accepts a Position object, which contains the
// current GPS coordinates
//
var onSuccess = function(position) {
    alert('Latitude: '      + position.coords.latitude      + '\n' +
          'Longitude: '     + position.coords.longitude     + '\n' +
          'Altitude: '      + position.coords.altitude      + '\n' +
          'Accuracy: '      + position.coords.accuracy      + '\n' +
          'Altitude Accuracy: ' + position.coords.altitudeAccuracy + '\n' +
          'Heading: '       + position.coords.heading       + '\n' +
          'Speed: '         + position.coords.speed         + '\n' +
          'Timestamp: '     + position.timestamp           + '\n');
};

// onError Callback receives a PositionError object
//
function onError(error) {
    alert('code: '      + error.code      + '\n' +
          'message: '   + error.message   + '\n');
}

navigator.geolocation.getCurrentPosition(onSuccess, onError);
```

Ajoutez votre code natif ...

- Possibilité de développer des plugins natifs
- Appels depuis le code Javascript :

```
cordova.exec(function(winParam) {}, function(error) {}, "service",  
            "action", ["firstArgument", "secondArgument", 42,  
            false]);
```

- winParam : Fonction appelée en cas de réussite
- Error : Fonction appelée en cas d'erreur
- Service : classe de la méthode appelée
- Action : méthode appelée

Utilisation d'un plugin

- Déclarer le plugin dans le fichier config.xml

```
<plugin name="<service_name>" value="<full_name_including_namespace>"/>
```

```
window.echo = function(str, callback) {  
    cordova.exec(callback, function(err) {  
        callback('Nothing to echo.');    }, "Echo", "echo", [str]);  
};
```

On définit la fonction echo sur l'objet window qui appelle la méthode echo de la classe Echo

```
window.echo("echome", function(echoValue) {  
    alert(echoValue == "echome"); // should alert true.  
});
```

Anatomie d'un plugin Android

```
package org.apache.cordova.plugin;

import org.apache.cordova.api.CordovaPlugin;
import org.apache.cordova.api.PluginResult;
import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

/**
 * This class echoes a string called from JavaScript.
 */
public class Echo extends CordovaPlugin {
    @Override
    public boolean execute(String action, JSONArray args, CallbackContext callbackContext) throws
    JSONException {
        if (action.equals("echo")) {
            String message = args.getString(0);
            this.echo(message);
            return true;
        }
        return false;
    }

    private void echo(String message, CallbackContext callbackContext) {
        if (message != null && message.length() > 0) {
            callbackContext.success(message);
        } else {
            callbackContext.error("Expected one non-empty string argument.");
        }
    }
}
```

Anatomie d'un plugin iOS

```
/****** Echo.h Cordova Plugin Header *****/  
  
#import <Cordova/CDV.h>  
  
@interface Echo : CDVPlugin  
  
- (void)echo:(CDVInvokedUrlCommand*)command;  
  
@end  
  
/****** Echo.m Cordova Plugin Implementation *****/  
  
#import "Echo.h"  
#import <Cordova/CDV.h>  
  
@implementation Echo  
  
- (void)echo:(CDVInvokedUrlCommand*)command  
{  
    CDVPluginResult* pluginResult = nil;  
    NSString* echo = [command.arguments objectAtIndex:0];  
  
    if (echo != nil && [echo length] > 0) {  
        pluginResult = [CDVPluginResult resultWithStatus:CDVCommandStatus_OK messageAsString:echo];  
    } else {  
        pluginResult = [CDVPluginResult resultWithStatus:CDVCommandStatus_ERROR];  
    }  
  
    [self.commandDelegate sendPluginResult:pluginResult callbackId:command.callbackId];  
}  
  
@end
```

Pour aller plus loin ...

- Possibilité d'inclure des bibliothèques Javascript spécifiques
- Possibilité d'écrire des plugins natifs
 - Si cela ne vaut pas un développement spécifique ...
- L'API en ligne : <http://docs.phonegap.com>
- Le Wiki : <https://github.com/phonegap/phonegap/wiki>

Le look d'une application native ?

- Malgré tous ces avantages
- Malgré tous ces composants
- Cela garde le look d'un site mobile et non d'une application :/

→ Des librairies peuvent vous y aider ...





jQuery Mobile

Sites mobiles cross platform

Petites questions ...

Qu'est-ce que jQuery ?

A quoi cela sert-il ?

Pourquoi l'utiliser ?

jQuery mais en mobile

- jQuery est utilisable sur Mobile
- jQueryMobile apporte des fonctionnalités cross platform
=> Layout, Dialog, Transitions, Multipage



- Différents thèmes
- Outil WYSIWYG de mise en forme
- Possibilité de décliner un site en 3 versions : classique, tablette et mobile



Avec Phonegap ?

- Intégration possible avec Phonegap
- Inclusion simple du code JS et HTML dans l'application et le tour est joué
- Quelques variables à régler

Petit tour des fonctionnalités



<http://demos.jquerymobile.com/1.4.4/>



Onsen UI

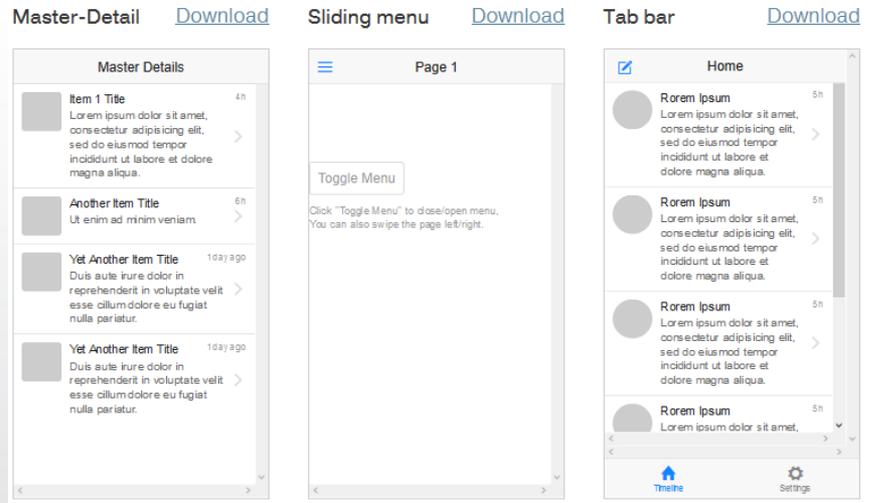
Fait pour Phonegap

Construit pour Phonegap

- Librairie JS
- Définit un lot de balises HTML et d'attributs sémantiques
- Fonctionnalités cross platform et widgets
=> Layout, Dialog, Transitions, Multipage

- Différents thèmes
- Thèmes et gabarits disponibles

→ <http://onsen.io/>

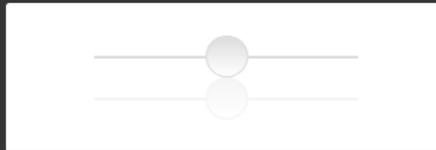


Construit pour Phonegap

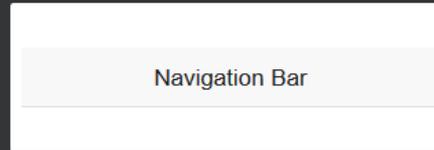
■ Switch



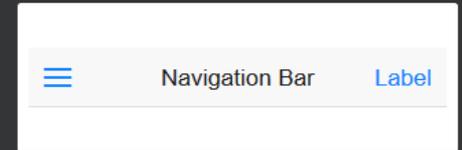
■ Range



■ Navigation Bar



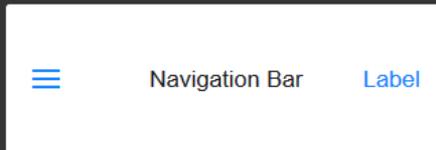
■ Navigation Bar Item



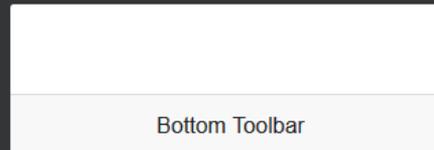
■ Navigation Bar with Outline Button



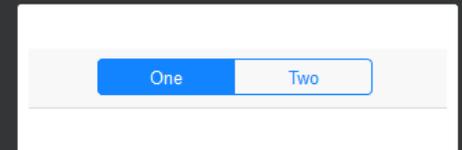
■ Transparent Navigation Bar



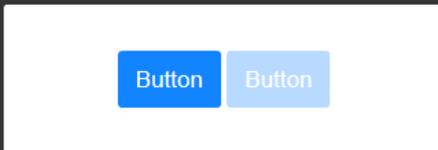
■ Bottom Bar



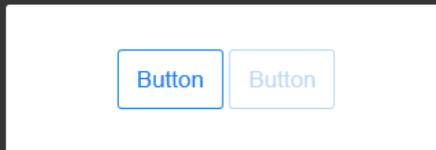
■ Navigation Bar with Segment



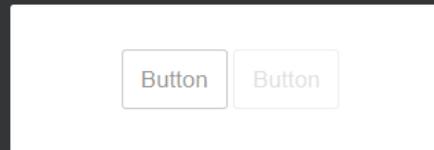
■ Button



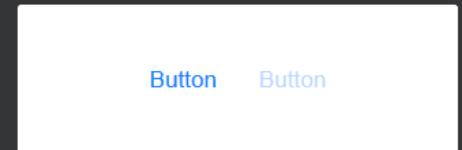
■ Outline Button



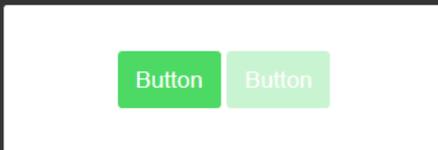
■ Light Button



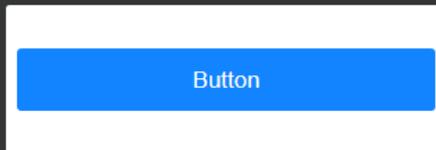
■ Quiet Button



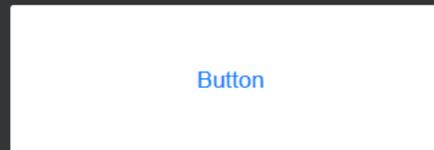
■ Call To Action Button



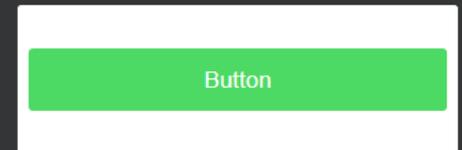
■ Large Button



■ Large Quiet Button



■ Large Call To Action Button



Avec Phonegap ?

- Une fois cordova installé

```
$ sudo npm install -g cordova
```

- Téléchargez un « thème »

```
$ cd /path/to/the/project  
$ cordova platform add ios (or android)
```

- Et démarrez!

```
$ cordova emulate
```

Structure d'un fichier

```
<!doctype html>
<meta charset="utf-8">
<link rel="stylesheet" href="lib/onsen/css/onsenui.css">
<link rel="stylesheet" href="lib/onsen/css/onsen-css-components.css">
<script src="lib/onsen/js/angular/angular.js"></script>
<script src="lib/onsen/js/onsenui.js"></script>
<script>
  ons.bootstrap();
  ons.ready(function() {
    // Add another Onsen UI element
    var content = document.getElementById("#my-content");
    content.innerHTML="<ons-button>Another Button</ons-button>";
    ons.compile(content);
  });
</script>
<body>
  <ons-navigator title="Navigator" var="myNavigator">
    <ons-page>
      <ons-button onclick="myNavigator.pushPage('page2.html')">Next Page</ons-button>
      <div id="my-content"></div>
    </ons-page>
  </ons-navigator>
</body>
```

Petit tour des composants



<http://components.onsen.io/>



Conclusion

All good things must come to an end

En bref ...

- Bien utilisé, Phonegap pourra vous faire gagner du temps en développement pour plusieurs cibles
- jQuery Mobile / OnsenUI vous permettra de faire de beaux sites / applicationscross-platform
- N'évite pas les tests sur les différentes plateformes

➔ Que doit-on faire avec notre appli ?



Questions

Et réponses ...



Coordonnées

Christophe BONNET

cbonnet@i2n.mc

+377 97 98 36 98

<http://www.i2n.mc>