



# PhoneGap

Build amazing mobile apps powered  
by open web tech.

# PhoneGap产生背景

- 1.移动互联网技术的迅猛发展，iOS、Android、WP7、BlackBerry、Symbian等各种OS（操作系统平台五花八门，开发工具层出不穷并专门化，开发语言存在较大差异。
- 2.一名iOS程序员无法忍耐Object-C生硬而又陌生的语法，他恰巧又注意到了Web巨大的前景，于是他认识到世界需要一种中间件，让熟知的HTML,CSS,JavaScript能够简单地部署在移动设备上


PhoneGap

# PhoneGap的特色

- 1.PhoneGap是一款让开发者用普通Web技术编写出能轻松调用API接口和进入应用商店的HTML5应用开发平台，是唯一一个支持全平台的开源移动框架。其开发成本低，成本项最多为原生APP的1/5.
- 2.兼容性：iOS,Android,BlackBerry,Windows Phone,Symbian,bada以及Web OS.
- 3.标准化HTML5和JS
- 4.大众化移动互联网开发平台



# PhoneGap的小伙伴们

- 俗话说一个好汉三个帮，再好的框架也不能应付所有的需求，在许多时候光靠PhoneGap是不够的，这时候就需要一些外界帮助了
  - **1.Jquery Mobile** 基于HTML5的跨平台开发框架，上手简单很强大。
  - **2.Jquery Touch** 命名得益于其强大的事件响应机制，一个独立的库，以JS为核心
  - **3.JQ iPhone UI** 仅仅是一组UI支持库，但高度仿真了iPhone的原声UI，用于一些特殊目的。
- 

# PhoneGap又饱受争议

- 1.因为“慢”饱受诟病  
使用PhoneGap开发的应用会比原声SDK开发出来的应用慢很多。
- 2.因为“快”引发争议  
反对者认为其属于“旁门左道”，但用好它也不是一件简单的事。



# PhoneGap开发环境的搭建

- <http://phonegap.com/getstarted/>

## 4 Run your new app on your device

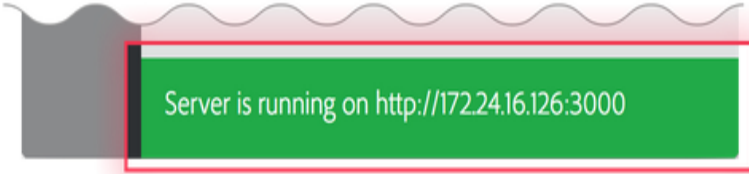
This is where things get real!

You'll need to connect the PhoneGap Developer app you installed in step 2 with the PhoneGap Desktop app you got running in step 3.

The PhoneGap Desktop app starts a small web server to host your project and once connected to that web server, the PhoneGap Developer app runs your project on your mobile device. Cool, right?


 The instructions to pair the Desktop and Developer apps are in our docs.

*PhoneGap Desktop: Server running*

A green notification box with a red border, containing the text "Server is running on http://172.24.16.126:3000".

Server is running on http://172.24.16.126:3000

*PhoneGap Developer App: Main screen*

A screenshot of the PhoneGap Developer App main screen. It features a dark background with a light blue "Connect" button. Below the button, there are instructions: "3-finger tap goes home" and "4-finger tap refreshes". A red box highlights the "Server Address" field which contains "172.24.16.126:3000".

> Server Address  
172.24.16.126:3000

Connect

3-finger tap goes home  
4-finger tap refreshes

# PhoneGap中的API能干啥？

- 虽然说使用PhoneGap主要是依靠HTML知识，但对于一名开发者来说最主要的还是PhoneGap中各种API的用法。
- **API的定义：** API（Application Programming Interface,应用程序编程接口）是一些预先定义的函数，目的是提供应用程序与开发人员基于某软件或硬件得以访问一组例程的能力，而又无需访问源码，或理解内部工作机制的细节。



# PhoneGap中常用API

- Accelerometer（加速度传感器）：Doodle jump
- Camera（摄像头）
- Capture（采集工具）：录音录像抓取图像
- ComPass（指南针）
- Contacts（联系人）
- Device（设备信息）：版本号、操作系统
- Events（系统时间）：电量过低、按键
- Geolocation（地理位置传感器）
- Media（媒体）
- Notification（提醒）





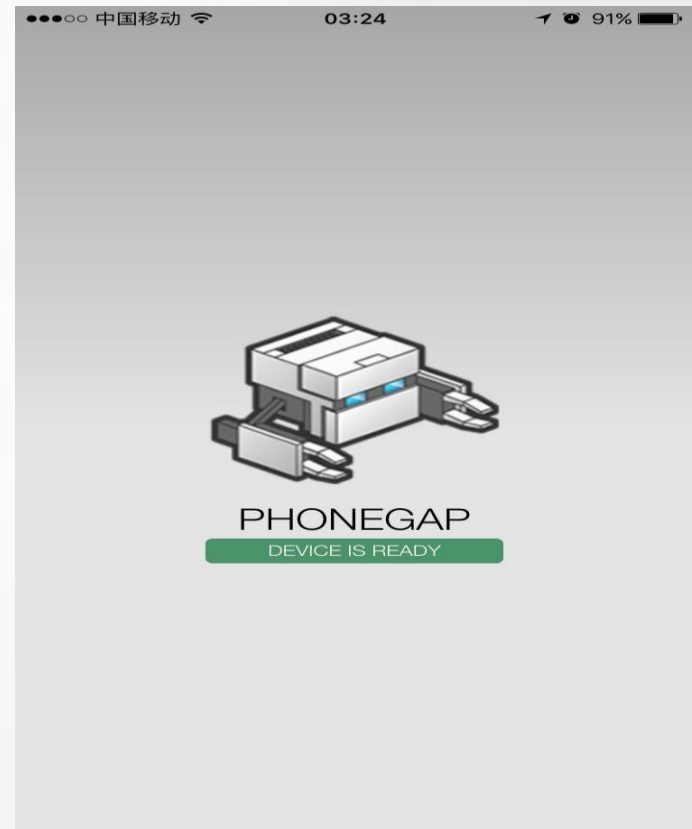
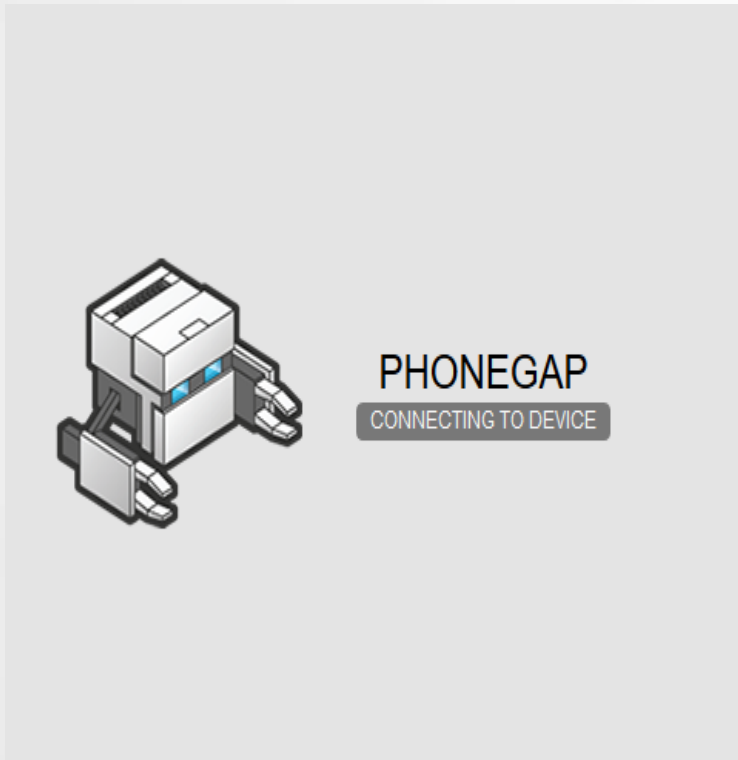
# 最简单的PhoneGap平台程序

```
<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="cordova.js"></script>
  <script type="text/javascript" src="js/index.js"></script>
  <script type="text/javascript">
    </script>
</body>
```


# 最简单的PhoneGap平台程序

```
document.addEventListener('deviceready', onDeviceReady, false);
function onDeviceReady()
|{
    receivedEvent('deviceready');//测试是否加载完成
}
function receivedEvent(id)
|{
    var parentElement = document.getElementById(id);
    var listeningElement = parentElement.querySelector('.listening');
    var receivedElement = parentElement.querySelector('.received');
    listeningElement.setAttribute('style', 'display:none;');
    receivedElement.setAttribute('style', 'display:block;');
    onsole.log('Received Event: ' + id);
}
```

# 最简单的PhoneGap平台程序



# 一、PhoneGap的生命周期与事件

- 生命周期：一个应用从开始运行—被手机加载—应用被退出之间的过程称为一个生命周期。
  - PhoneGap的生命周期可以分为15个事件：
    - 1.程序加载事件 deviceready、 pause、 resume.
    - 2.被动消息事件 online offline batterylow等
    - 3.主动消息事件backbutton、 volumedownbutton等
- 

# 常见事件

- **Deviceready**: 当设备加载完毕后会触发该事件
- **pause**: 当程序被暂停到后台运行触发
- **Resume**: 当程序被从后台激活到前台运行
- **Batterylow**: 当设备剩余电量低于某值时触发
- **Volumedownbutton**: 当用户按下“音量减小”按钮时触发



注意：PhoneGap有时不能“通吃”

如iOS系统手机不存在返回键，则需要靠返回键相应的事件在Android可以实现在iOS则实现不了



## 二、设备信息的获取

- PhoneGap的一个最大优势在于可以通过JS调用设备来访问设备上的硬件信息，从而实现一些原本只有依靠原生SDK才能达到的目的。



# 设备信息的获取

- 利用到了device的name、phonegap、platform属性等。
- Name---设备名，即使同款也可能不同，范例由于是虚拟机无法获取设备名称所以显示undefined。
- Phonegap—版本号
- Platform-操作系统
- Uuid-设备编号
- Version-具体版本号





## 三、通讯录信息的获取

- 1.创建一个contact对象  
PhoneGap把有关通讯录的信息封装在一个Contact类中，因此在对通讯录操作前应先创建一个contact对象。
- 2.利用find()方法查询通讯录
- 3.contact对象的一系列属性  
id、displayname、phonenumbers等
- 4.增删改查其他操作



## 四、PhoneGap的消息提示

- PhoneGap的notification类中封装了一系列设备视听触觉的通知，可以用于定制专门的消息推送等活动。
- 1.notification警告的使用  
相比alert(), notification.alert()具有更强大的定制型与易用性，因为他可以弹出可自定义的消息框。
- 2.notification.confirm确认对话框
- 3.notification显示可传参对话框 prompt()
- 4.notification控制蜂鸣器和震动

# 五、加速度传感器

- PhoneGap提供了accelerometer类用来接收来自加速度传感器的数据，从而监测到设备在空间上的位置变化。
- 1.获取当前加速度  
`accelerometer.getCurrentAcceleration()`
- 2.监视设备的加速度
  - (1)不停调用1.
  - (2)利用`watchAcceleration()`
- 3.应用： 测量运动量、跑酷游戏、摇一摇

## 六、设备传感器

- 除了加速度传感器，PhoneGap也提供了对其他设备传感器的调用方法。Geolocation与compass类
- 1.利用getCurrentposition()获取地理坐标
- 2.利用watchposition()监控位置变化
- 3.设备方向的获取getCurrentheading()
- 4.设备方向的监控watchcurrentheading()



# 七、PhoneGap对音频的控制

- 在PhoneGap中，Media对象提供了对音频文件的播放和录制功能。
- 1.利用PhoneGap播放音频的方法
- 2.利用pause()暂停播放音乐
- 3.利用stop()停止播放音乐
- 4.PhoneGap录制声音 startrecord()  
stoprecord()



八、多媒体资源的获取

九、PhoneGap的文件操作

十、PhoneGap本地存储的使用



实例：

打飞机手机版实现

