

HTML5plus 移动 App 开发入门

(v0.3)

更新时间：2014 年 8 月 4 日

HTML5 Plus 应用概述

HTML5 Plus 规范

通过 HTML5 开发移动 App 时，会发现 HTML5 很多能力不具备。为弥补 HTML5 能力的不足，在 W3C 中国的指导下成立了 www.html5plus.org 组织，推出 HTML5+规范。

HTML5+规范是一个开放规范，允许三方浏览器厂商或其他手机 runtime 制造商实现。

HTML5+扩展了 JavaScript 对象 plus，使得 js 可以调用各种浏览器无法实现或实现不佳的系统能力，设备能力如摄像头、陀螺仪、文件系统等，业务能力如上传下载、二维码、地图、支付、语音输入、消息推送等。

HBuilder 的手机原生能力调用分 2 个层面：

- 跨手机平台的能力调用都在 HTML5+规范里，比如二维码、语音输入，使用 `plus.barcode` 和 `plus.speech`。编写一次，可跨平台运行。
- Native.js 是另一项创新技术。手机 OS 的原生 API 有四十多万，大量的 API 无法被 HTML5 使用。Native.js 把几十万原生 API 封装成了 js 对象，通过 js 可以直接调 ios 和 android 的原生 API。这部分就不再跨平台，写法分别是 `plus.ios` 和 `plus.android`，比如调 ios game center，或在 android 手机桌面创建快捷方式。

Native.js 的用法示例，`var obj= plus.android.import("android.content.Intent");`，将一个原生对象 `android.content.Intent` 映射为 js 对象 `obj`，然后在 js 里操作 `obj` 对象的方法属性就可以了。

Native.js 的详细教程可以参考：<http://download.dcloud.net.cn/HTML5%2B应用开发-Native.js.pdf>

HTML5+ App

使用 HTML5+开发的移动 App 并非 mobile web 页面。这是新手最容易混淆的地方。mobile web 的文件存放在 web 服务器上，而移动 App 的文件存放在手机本地，编写移动 App 的 html、js、css 文件被打包到 ipa 或 apk 等原生安装包，在手机客户端运行。

当然这些移动 App 里某些页面也可以继续从服务器端以网页方式下行。

所以 mobile web，在 HBuilder 里新建项目时，属于 web 项目。不要放置到移动 App 项目下。mobile web 项目也不能真机联调和打包。

举几个例子。

例 1：一个 mobile web 项目，想打包成移动 App。

- 在 HBuilder 里新建一个 web 项目，把 mobile web 代码放进去。
- 在 HBuilder 里新建移动 App
- 在新建的移动 App 下找到 `manifest.json`，将其中的入口页面配置为 mobile web 的网络地址。
- 然后点发行打包，就得到一个移动 App 的安装包。除了可发行到 Appstore 和桌面有个快捷方式外，与浏览器的体验不会有其他区别。
- 另外其实 mobile web 的代码，也可以判断自己运行的环境，如果 UA 里包含

“Html5Plus”，也可以写 plus 对象来调用原生能力。

例子 2：正规的移动 App（没有网页进度条）

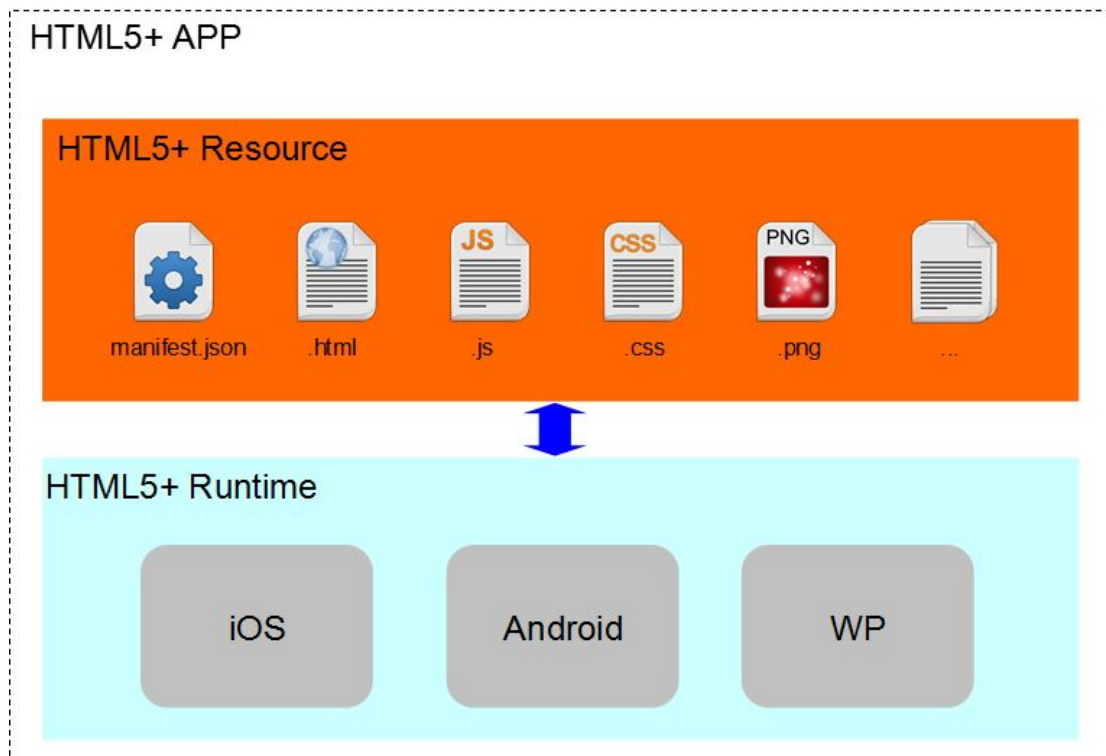
- 在 HBuilder 里新建移动 App 项目
- 在移动 App 里编写 html、Js、css 文件，本地 js 通过 ajax 方式请求服务器数据，通过 plus.net 对象避开跨域限制。
- 移动 App 里的 js 可以通过 plus 对象调用手机原生能力
- 编写好的移动 App 点打包变成安装包。

例子 3：混合型移动 App

这里的混合型移动 App，所指并非是原生和 HTML5 的 hybrid App，而是指一部分页面是本地的 HTML，通过 ajax 与服务器交互，另一部分页面是从服务器下行的 mobile web 页面。

- 分别新建一个 web 项目和一个移动 App 项目
- 在移动 App 里的某个 html 里通过 <a href= 或者 location.href=指定 mobile web 的页面地址。

HTML5+ 应用架构



HTML5+ 规范 API 及 demo 示例

最新规范请参考 <http://www.html5plus.org/#specification>

手机端体验各个 API 的实现效果，ios 手机在 Appstore 搜索 Hello H5+，Android 手机下载 <http://dcloud.io/helloh5/HelloH5.apk>。

在 HBuilder 中新建移动 App，选 Hello H5+，即可看到这个 demo 的源代码。

开发环境 HBuilder

HBuilder 内置 HTML5+ APP 开发环境，提供一套完整的移动应用开发解决方案。内置 HTML5+ API 语法提示，提高开发效率；集成真机运行环境，方便开发后即时在真机上查看运行效果；集成应用云端打包系统，不用部署 xcode 和 Android sdk 就可以打包应用。使开发者只需要使用 HTML5、CSS、CSS 技术就可以快速开发跨平台的移动应用。

下载地址：<http://www.dcloud.net.cn/>

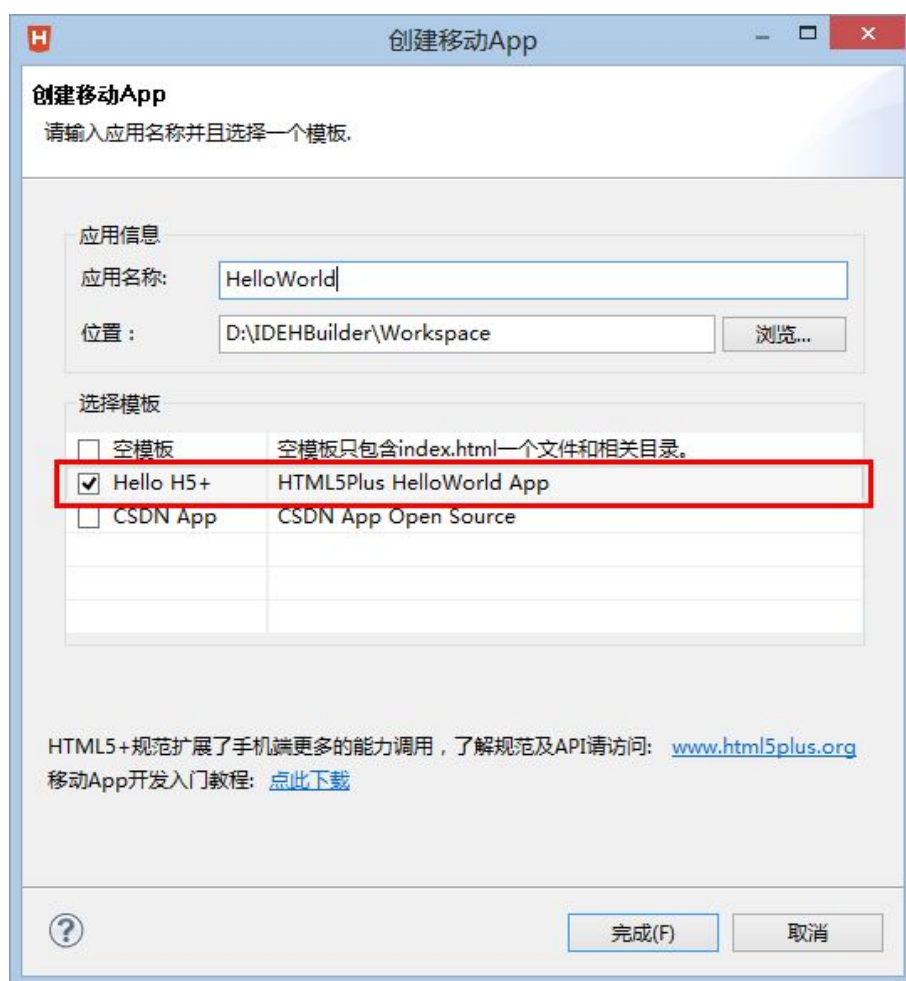
平台支持

- iOS 5.0 及以上
- Android 2.3 及以上
- WP 8.0 及以上（开发中）

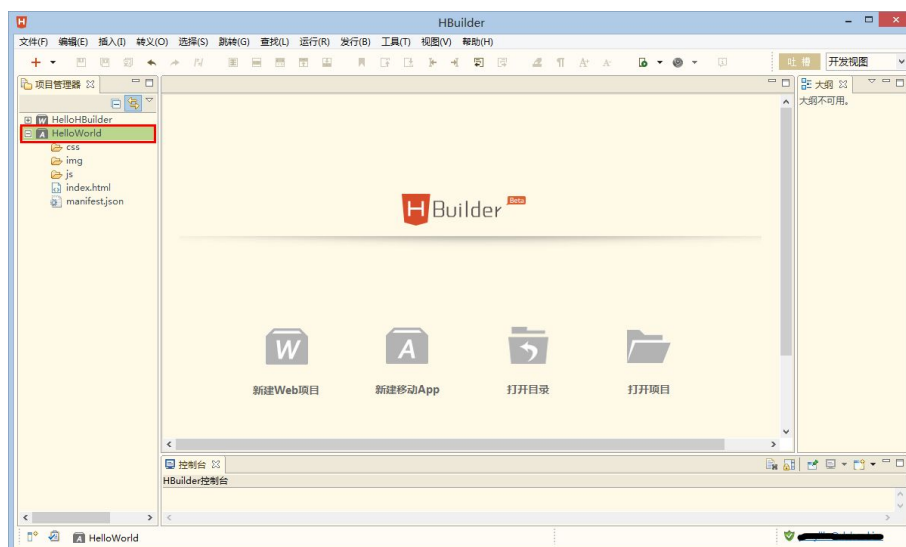
手把手教你开发 HelloWorld

创建 HelloWorld 应用

- 启动 HBuilder；
- 在菜单栏中选择“文件”->“新建”->“移动 App”（快捷键 Ctrl+N A），打开“创建移动 App”对话框，在应用名称中输入“HelloWorld”；
注意新建移动 App 需要联网分配一个 appid，在真机联调、打包发行时都需要这个 ID，所以不联网无法创建移动 App。

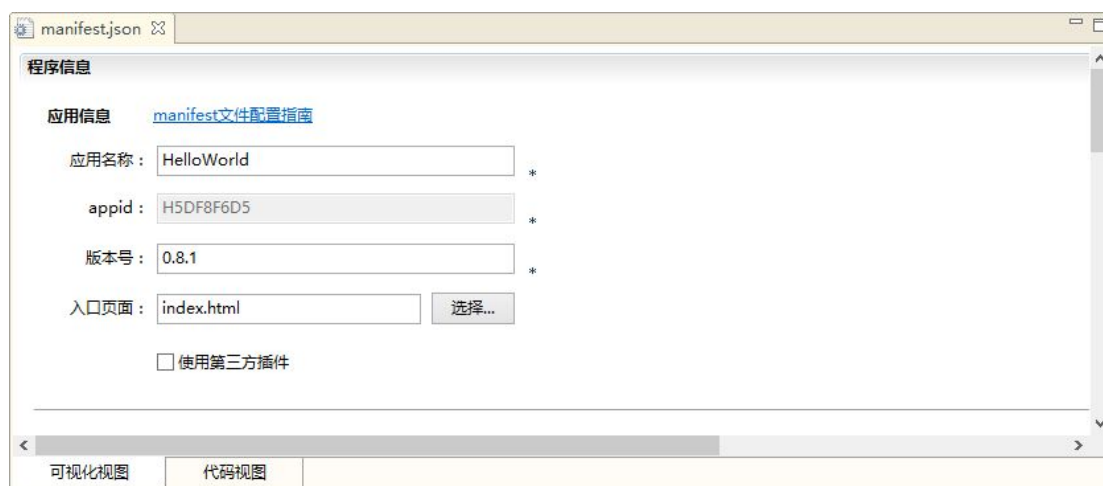


- 创建完成后，会在项目管理器中显示新建的“HelloWorld”项目



manifest.json

在项目管理器中双击“manifest.json”文件，打开应用配置页面：



对于要打包的原生应用而言，其各种配置均在此处。具体配置教程见：<http://download.dcloud.net.cn/移动应用manifest文件配置指南.pdf>，或者点击配置页面上的“manifest 文件配置指南”链接。

调用 HTML5+ API

在项目管理器中双击“index.html”文件，对于 HTML5+ 应用的页面有一个很重要的“plusready”事件，此事件会在页面加载后自动触发，表示所有 HTML5+ API 可以使用，在此事件触发之前不能调用 HTML5+ API，所以应该在此事件回调函数中调用页面初始化需要调用的 HTML5+ API，而不应该在 onload 事件中调用：

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8"/>
  <meta name="viewport" content="initial-scale=1.0, maximum-scale=1.0,
user-scalable=no"/>
  <title>Hello world</title>
  <script type="text/javascript">
// 扩展 API 是否准备，如果没有则监听 “plusready”事件
if(window.plus){
  plusReady();
}else{
  document.addEventListener( "plusready", plusReady, false );
}
// 扩展 API 准备完成后要执行的操作
function plusReady(){
  var self = plus.webview.currentWebview();
  // ... code
}
  </script>
</head>
<body>
```

编辑程序启动后默认显示的页面 index.html，在页面中添加一个按钮，点击后将打开新页面加载“<http://m.csdn.net/>”，为了实现此功能，我们需要用到 HTML5+扩展 API 中 plus.webview.currentWebview()方法创建窗口：

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="utf-8"/>
5   <meta name="viewport" content="initial-scale=1.0, maximum-scale=1.0, user-sca
6   <title>Hello world</title>
7   <script type="text/javascript">
8   // 扩展API是否准备, 如果没有则监听“plusready”事件
9   if(window.plus){
10    plusReady();
11   }else{
12    document.addEventListener( "plusready", plusReady, false );
13   }
14   // 扩展API准备完成后要执行的操作
15   function plusReady(){
16    var self = plus.webview.currentWebview();
17    // ... code
18   }
19   // 打开新Webview窗口
20   function openNewWebview(){
21    var url="http://m.csdn.net/";
22    plus.webview.create(url).show();
23   }
24   </script>
25 </head>
26 <body>
27   Hello world<br/>
28   <button onclick="openNewWebview()">打开新页面</button>
29 </body>
30 </html>

```

编辑完成后，按 Ctrl+S 键保存。

真机运行

添加调用 HTML5+API 代码后，我们可以通过真机运行来查看效果，

在 HBuilder 更改页面并保存后，可立即同步在真机上看到保存后的显示效果。同时页面做的错误以及 console.log，也可以在真机运行时从手机端反馈回到 HBuilder 的控制台。

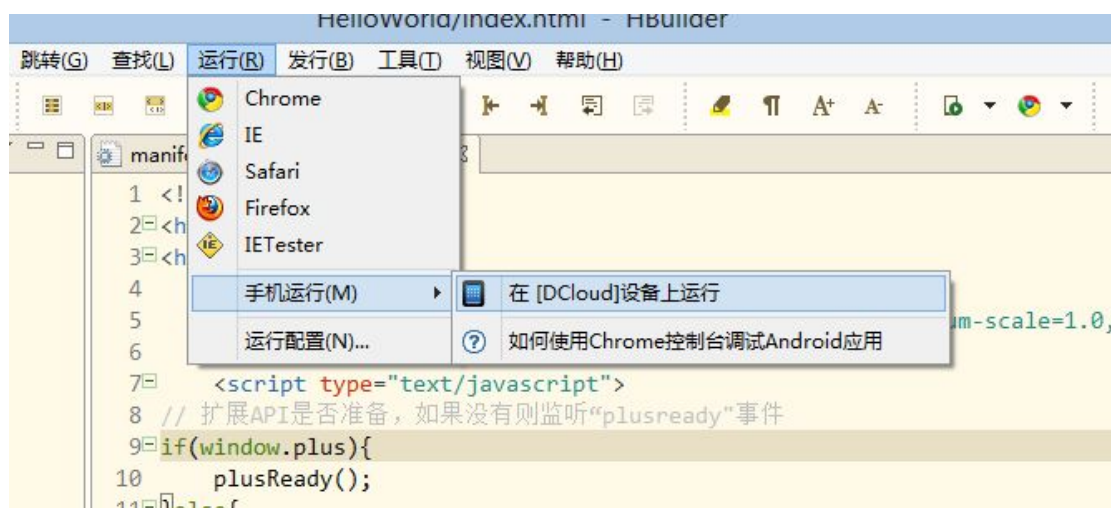
注意只有移动 App 项目才可以真机联调。

iOS 真机运行

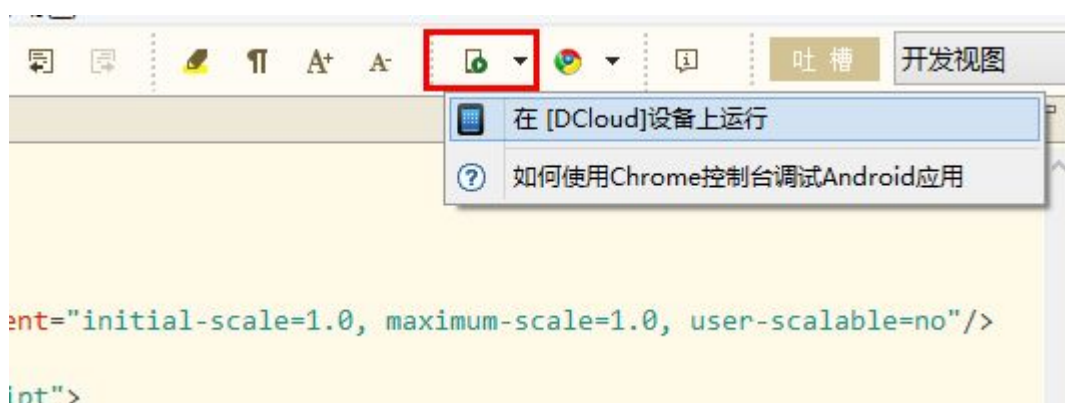
在 HBuilder 的“项目管理器”中选择创建的“HelloWorld”应用。

启动真机运行

将 iOS 设备连接到电脑，这时 HBuilder 会自动检测连接到电脑上的设备，通过菜单栏中的“运行”菜单启动：



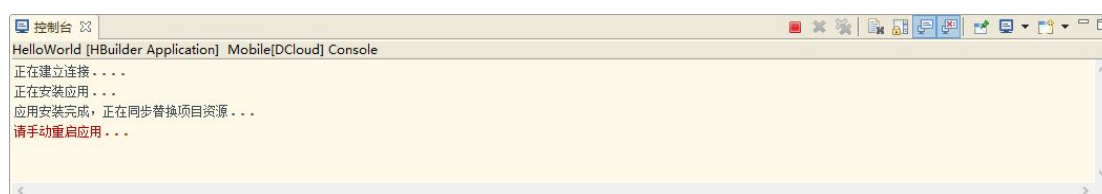
也可通过工具栏启动：



注：如果没有发现设备，请确认电脑是否安装 iTunes 程序并更新到最新版本？或者尝试拔下数据线重新连接！

安装真机运行环境

启动真机运行后，HBuilder 会自动在设备上安装真机运行环境，并在控制台中显示以下信息：

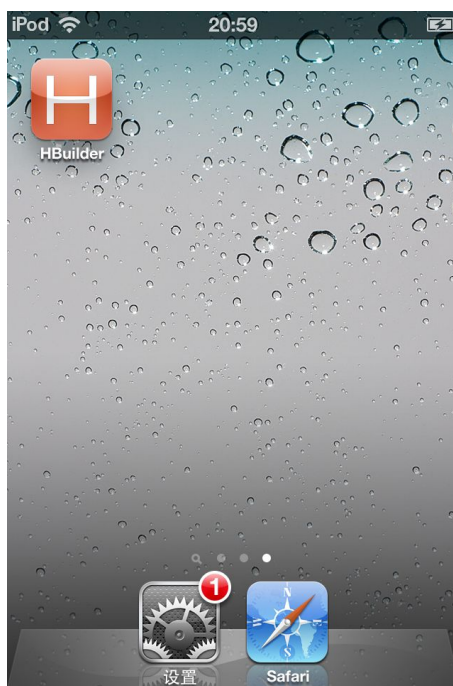


注：如果提示错误信息，请尝试点击“终止”按钮后重新启动真机运行！

真机运行应用

在 iOS 设备上可看到新安装的 HBuilder 应用图标，手动点击运行

注意：真机联调 App 时，提供的是一个测试环境，并不真实发生打包，手机端 App 界面还有 develop 的水印，其手机桌面显示名字为 HBuilder，图标也是 HBuilder 的图标，此时并不是开发者真实 App 的名字图标。正式的 App 需要通过发行-打包来制作，打包后的 App 名字图标都是开发者定义的，里面也没有 develop 的水印。

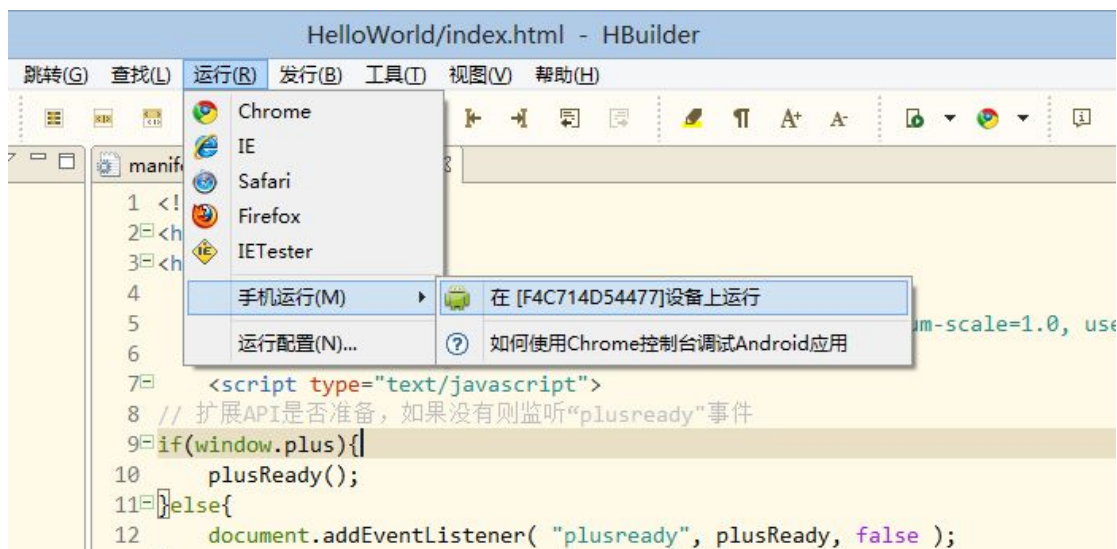


启动后会弹出提示框，选择“确定”，显示以下页面：

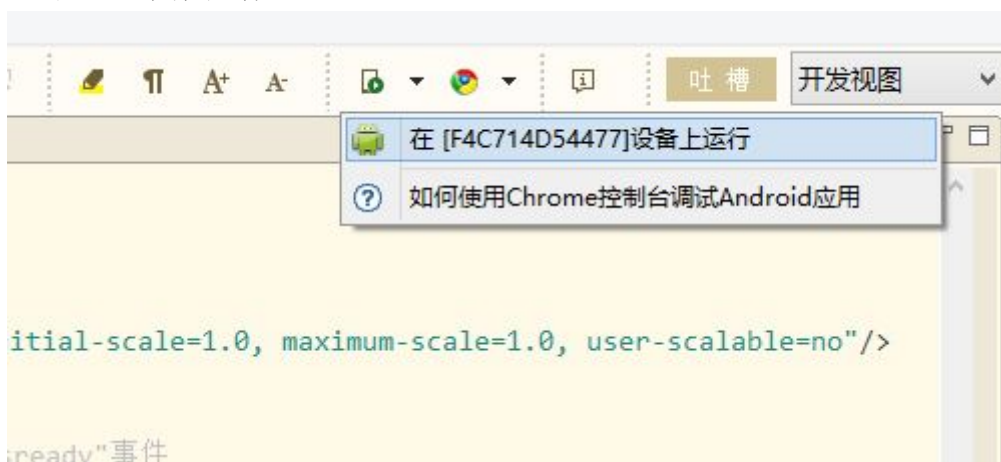


Android 真机运行

将 Android 设备连接到电脑，这时 HBuilder 会自动检测连接到电脑上的设备，通过菜单栏中的“运行”菜单启动：



也可通过工具栏启动：

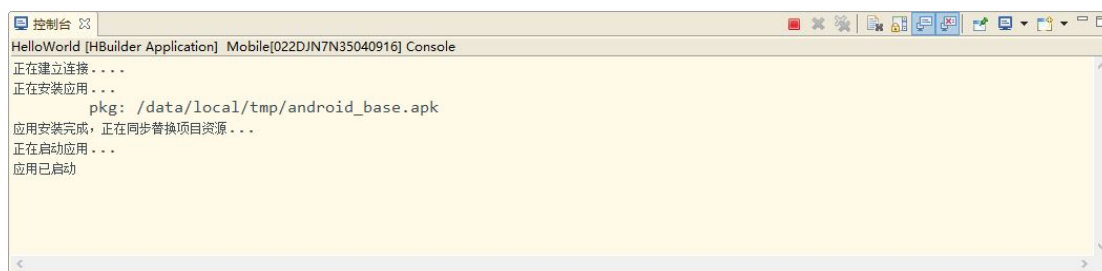


注：如果没有发现设备，请安装“360 手机助手”等手机助手工具安装设备的驱动，保证手机和 PC 可以连接。也可以尝试拔掉手机数据线重新连接，另外有些数据线质量不好，会无法稳定连接。

运行一次后，关闭时务必点停止运行的红色方块按钮，否则下次连接很可能阻塞。

安装真机运行环境

启动真机运行后，HBuilder 会自动在设备上安装真机运行环境，并在控制台中显示以下信息：



注：如果提示错误信息，请尝试“终止”后重新启动真机运行！

真机运行应用

在 Android 设备会自动安装 HBuilder 应用，在桌面上可看到新安装的 HBuilder 应用图标（有些设备会在应用列表页面中）：

注意：真机联调 App 时，提供的是一个测试环境，并不真实发生打包，手机端 App 界面还有 develop 的水印，其手机桌面显示名字为 HBuilder，图标也是 HBuilder 的图标，此时并不是开发者真实 App 的名字图标。正式的 App 需要通过发行-打包来制作，打包后的 App 名字图标都是开发者定义的，里面也没有 develop 的水印。



启动后会弹出提示框，选择“确定”，显示以下页面：



部署发布

完成应用页面的编辑后，需要正式发布到个大市场，这时需要从云平台进行安装包的制作。通过菜单栏中的“发行”->“App 打包”，打开“App 云端打包”对话框提交。

注意只有移动 App 项目才可以打包。

iOS 发布

对于 iOS 平台，可以选择越狱包或正式包（Appstore 专用），前者只能安装在已越狱的设备上，后者则可通过 iDP 证书打包提交到 Appstore 发布、或通过 iEP 证书打包在企业内部发布。

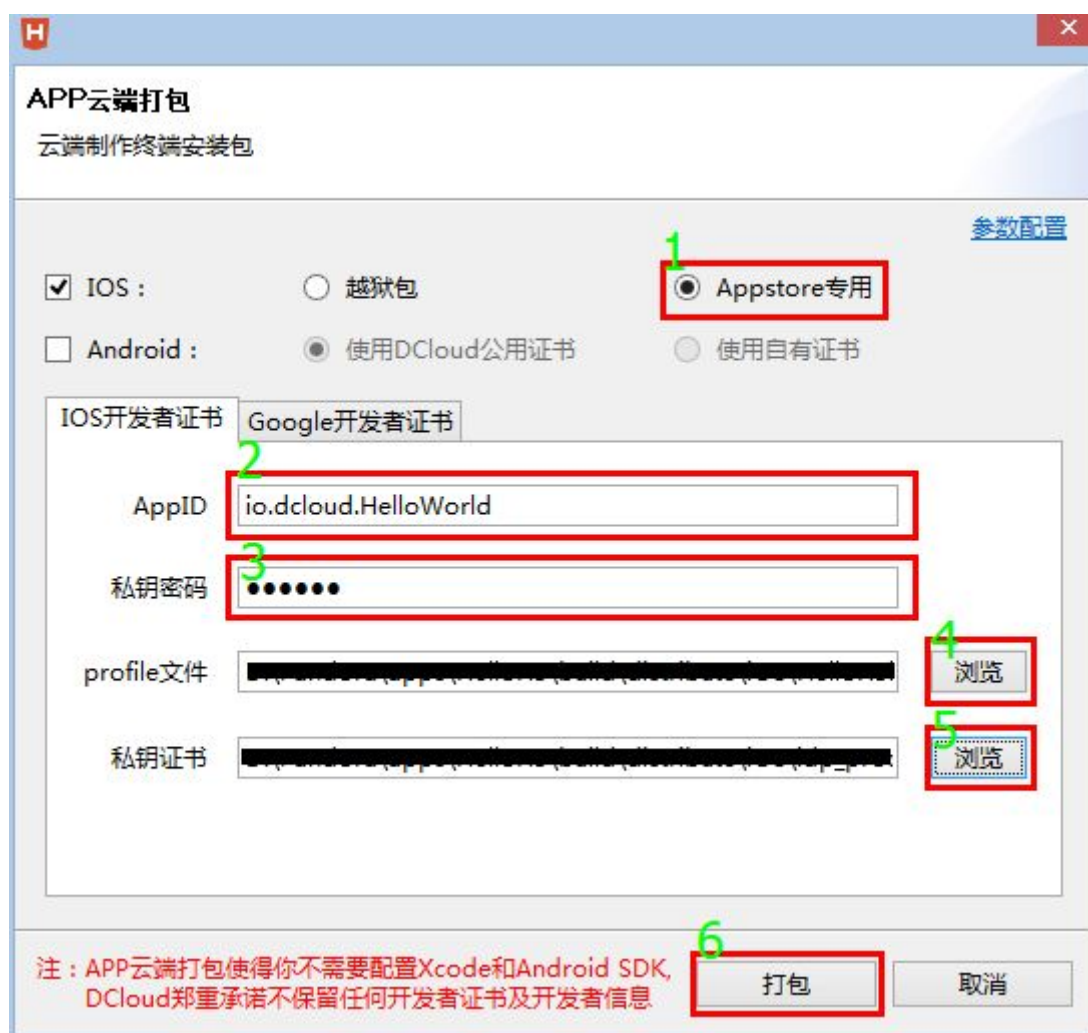
配置打包信息

- 越狱包



AppID: iOS 应用标识，推荐使用反向域名风格的字符串，如“com.domainname.appname”。

- 正式包



AppID: iOS 应用标识，推荐使用反向域名风格的字符串，如“com.domainname.appname”，必须与 profile 文件绑定的 App ID 匹配。

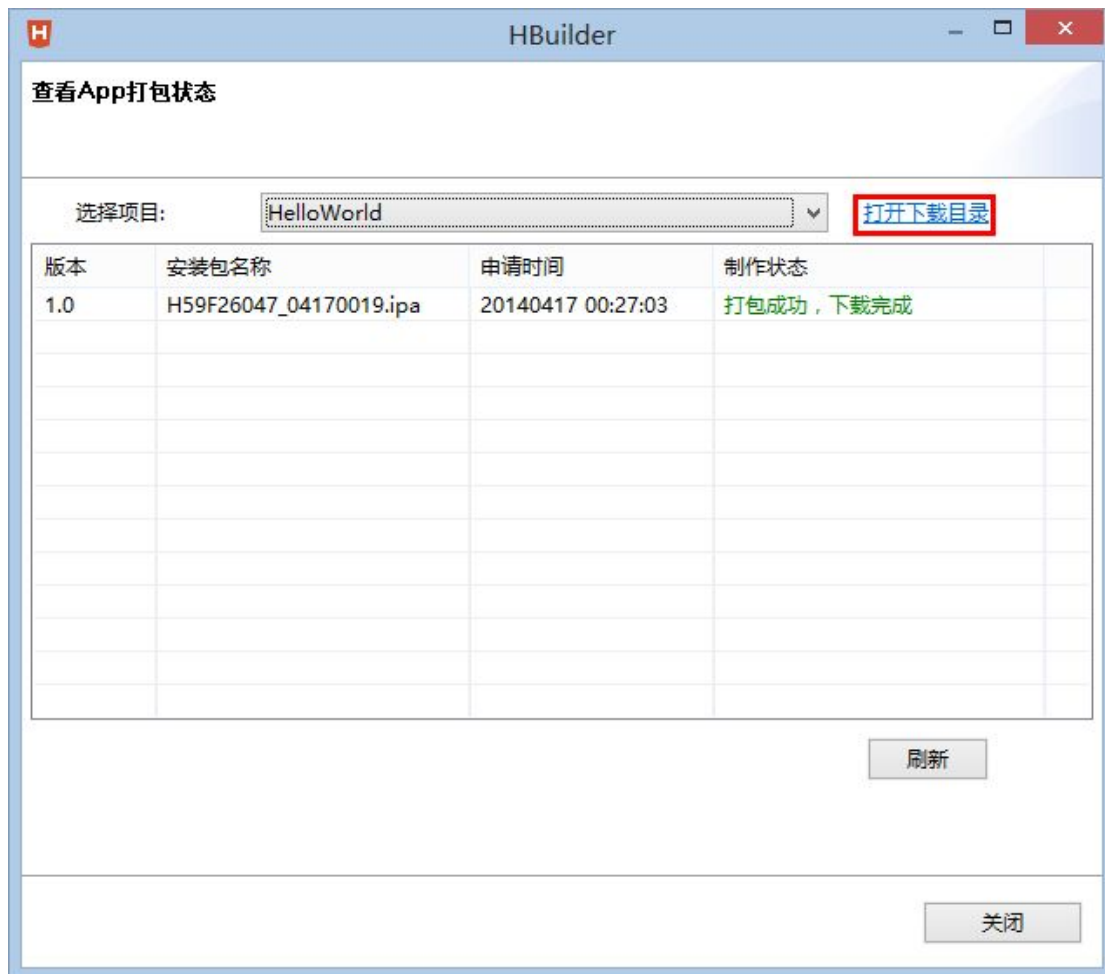
私钥证书: iOS Certificates 文件（.p12）；

私钥密码: 导入私钥证书的密码；

Profile 文件: iOS Provisioning Profile 文件（.mobileprovision），必须与 App ID 和私钥证书匹配；

查看打包状态

通过菜单栏中的“发行”->“查看打包状态”，打开“查看 App 打包状态”对话框，可查看打包历史记录和状态：



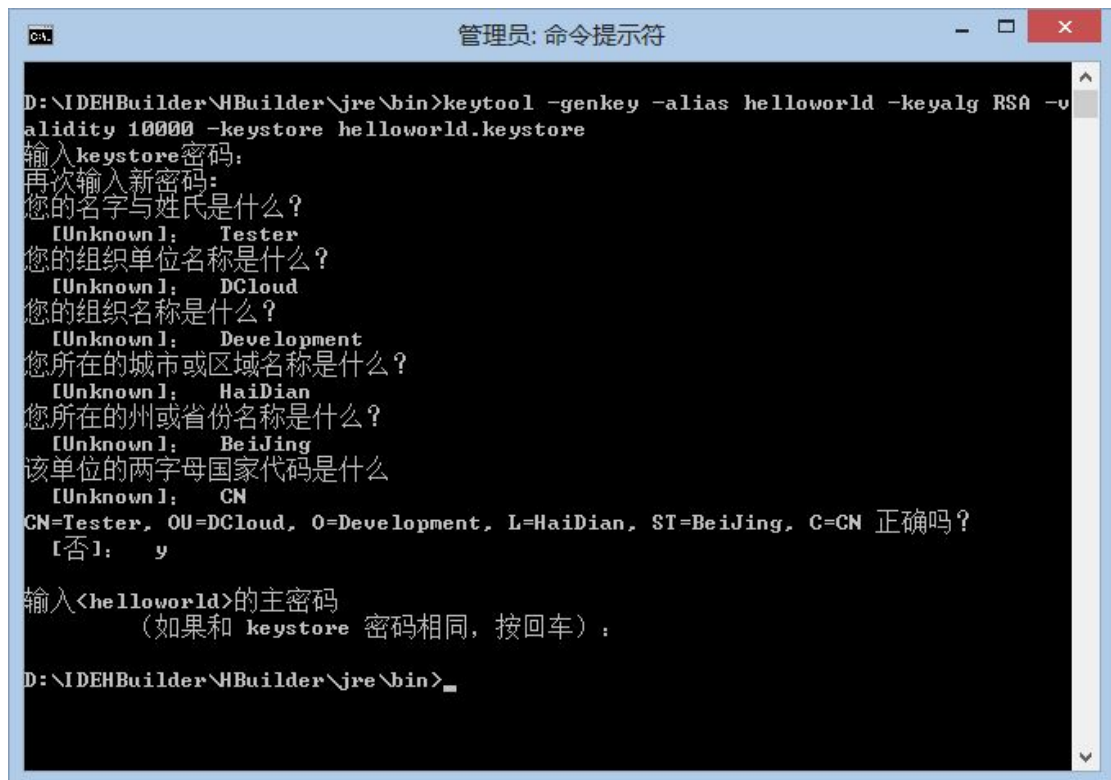
如果“制作状态”栏显示“打包成功，下载完成”则表示云端打包完成，可点击“打开下载目录”查看下载的安装包。

Android 发布

对于 Android 平台，可以选择使用 DCloud 生成的公用证书或自己生成的证书，两者不影响安装包的发布，唯一的差别就是证书中开发者和企业信息不同。

生成 Android 签名证书

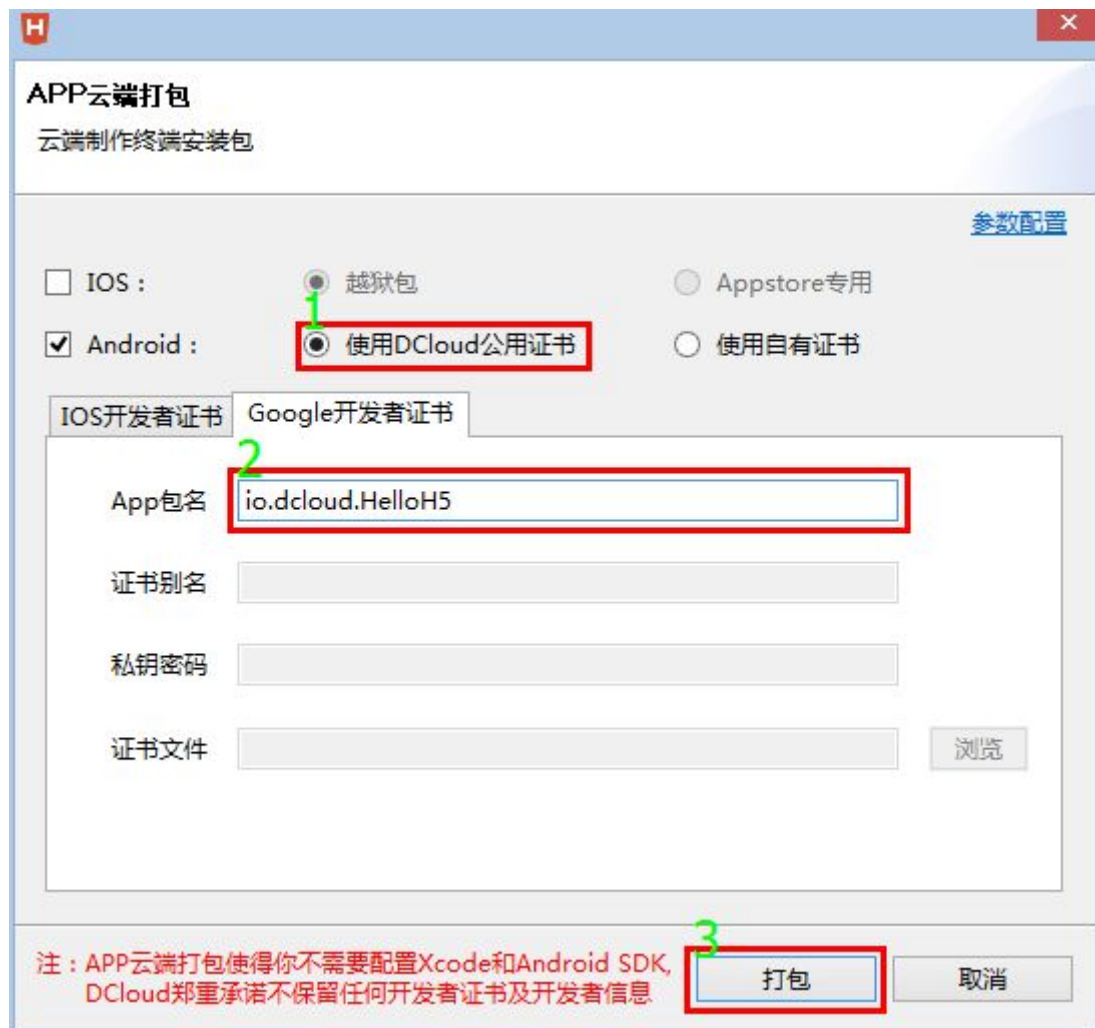
（使用 DCloud 公用证书可忽略）确保电脑上安装了 JRE，我们将使用 JRE 自带的创建和管理数字证书的工具 **Keytool**。使用以下命令生成证书：



- keystore helloworld.keystore 表示生成的证书，可以加上路径（默认在用户主目录下）；
- alias helloworld 表示证书的别名是 helloworld；
- keyalg RSA 表示采用的 RSA 算法；
- validity 10000 表示证书的有效期是 10000 天。

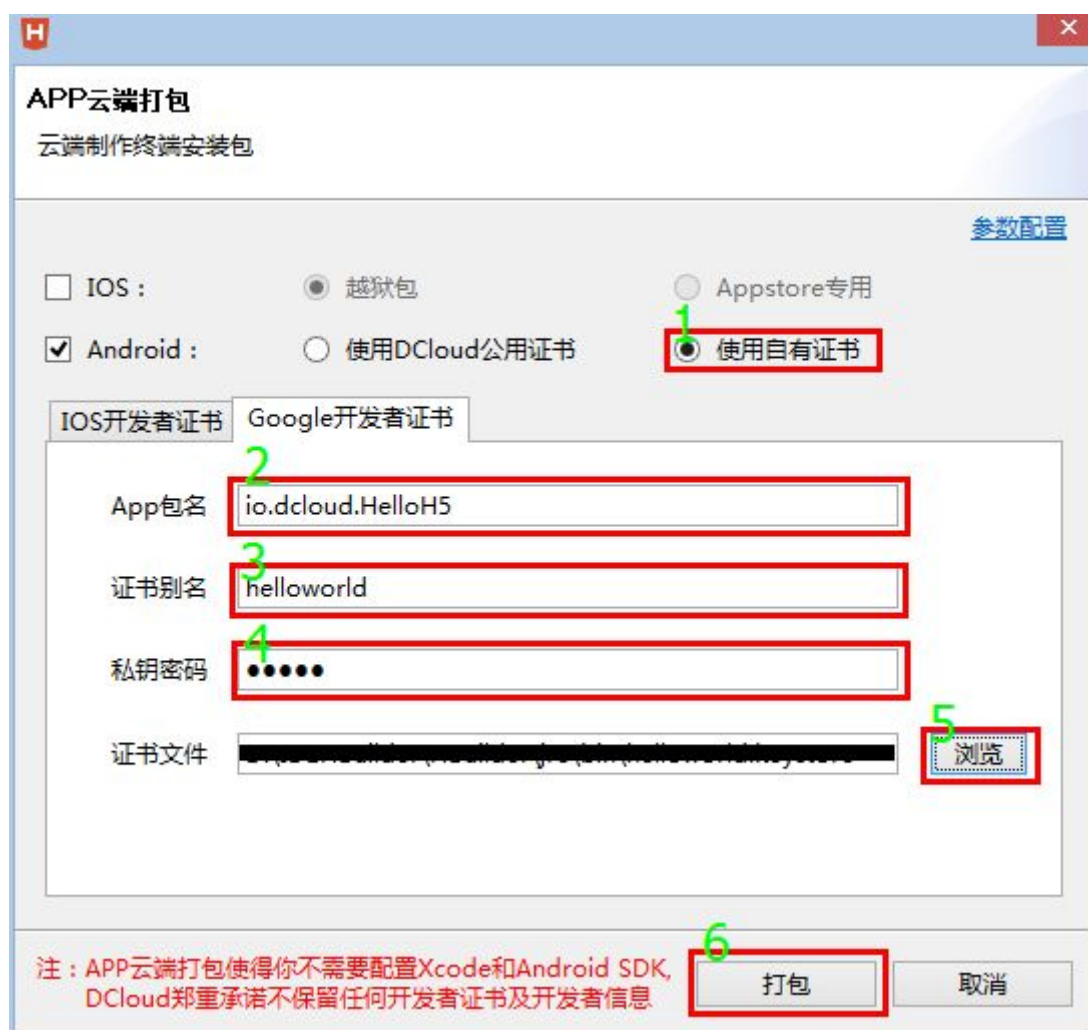
配置打包信息

- 使用 DCloud 公用证书



App 包名： Android 应用包名，推荐使用反向域名风格的字符串，如“com.domainname.appname”。

- 使用自有证书



App 包名： Android 应用包名，推荐使用反向域名风格的字符串，如“com.domainname.appname”。

证书别名： 生成证书时使用 -alias 参数设置的证书别名；

私钥密码： 生成证书时使用的 keystore 密码；

证书文件： 生成证书时使用 -keystore 参数设置的证书保存路径；

查看打包状态

通过菜单栏中的“发行”->“查看打包状态”，打开“查看 App 打包状态”对话框，可查看打包历史记录和状态：

UI 框架

在 UI 方面，如 Sencha Touch、jQTouch、PhoneJS 等框架在一些低端 Android 机上无法流畅运行。Ratchet 是我们认为的一个不错的 UI 库，使用 css 来定义手机控件样式。该开源 UI 库的地址是：goratchet.com/components。开发者可以从这里扣取各种适合手机的控件的 css。但也仅限于使用 ratchet 的 css，因为它的 js 也不够高效。

调试、断点

除了真机运行，我们还可以利用 chrome 和 safari 的开发者控制台来调试 5+App。

可以使用真机插上数据线，也可以使用模拟器。所有 Api 包括 plus 的各种 api，甚至包括 plus.ios 和 plus.android 的原生对象，都可以调试。

在 HBuilder 的菜单运行 - 手机运行里有教程的链接，链接如下：

<http://dcloud.io/docs/debug.android/>

开发资源

API 参考：<http://www.html5plus.org/#specification>

Hello H5+，这个示例应用有几乎所有 plus api 的调用样例：

iOS：Appstore 中搜索 Hello H5+。

Android：访问 <http://www.dcloud.io/helloh5/HelloH5.apk>



获取 Hello H5+的源代码，在 HBuilder 中新建移动 App，选 Hello H5+。可以查看所有 plus api 的调用样例代码。

CSDN 官方资讯手机版，也是基于 5+开发的：

iOS：Appstore 中搜索 CSDN。

Android：访问 <http://d.dcloud.net.cn/csdn/csdn.apk>



FAQ

只能云端打包吗？支持本地打包吗？

支持本地打包，参考：

Android: <http://ask.dcloud.net.cn/article/38>

iOS: <http://ask.dcloud.net.cn/article/41>

js 不能跨域，本地 HTML5 如何与服务器交互？

使用 `plus.net.XMLHttpRequest` 对象可以跨域

plus 对象浏览器不支持，是不是需要引用什么 js 框架？

不需要引入 js 框架，plus 对象需要系统底层支持而不是单纯的一个 js 框架就能解决问题。

plus 对象在 DCloud 的 5+runtime 里已经实现。使用 HBuilder 打包时，是把工程里的 html、js、css 和 DCloud 的 5+runtime 混编打包成 ipa 或 apk 的。

也可以把 5+runtime 理解为 cef，一种没有地址栏的、实现了 plus 对象的增强浏览器内核。

plus 规范是开放的吗，我可以自己实现 plus 的 runtime 或在我自己的浏览器里支持 plus 对象吗？

plus 规范属于 HTML5Plus.org，是开放规范，任何公司或个人都可以基于 HTML5+ 规范开发自己的手机端实现。

什么是“响应式应用”：

一套代码根据不同运行环境调用不同能力来优化用户体验。

比如文本框，在普通浏览器下就是一个文本输入框，在 Chrome 浏览器下运行时加个判断调 Chrome 的语音输入，而运行在 App 模式下，则再加一个判断调用原生的语音识别 SDK，比如 `plus.speech`。

这样一个应用可以根据设备的不同而最大化的优化用户体验，这种模式很有魅力，对用户而言体验更好，对开发者而言成本更低。

iOS 如何安装越狱包

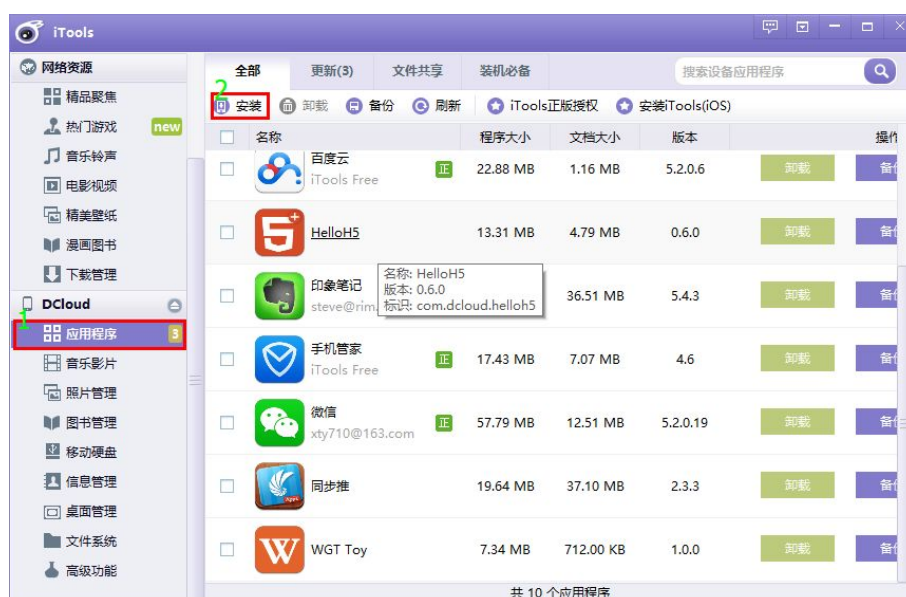
iOS 设备需要进行越狱操作（威锋网越狱教程）后才能安装越狱包，通常可以使用第三方工具进行管理，如 iTunes（[官网](#)）：

使用 iTunes 工具安装

- 安装 iTunes 工具并启动（注：需安装 [iTunes](#) 工具）
- 将 iOS 手机连接到电脑，这时 iTunes 工具会提示发现设备，并确定设备越狱状态为“已越狱”：



- 右侧设备下选择“应用程序”，点击“安装”选择越狱包 ipa 文件：



- 安装完成后即可在手机上打开运行

iOS 安装越狱包提示“验证失败”错误

iOS 设备需要进行越狱操作（威锋网越狱教程）后才能安装越狱包，对于 iOS7.x 系统越狱操作后需要确认是否安装安装 afc2add、AppSync 工具？

安装 afc2add、AppSync 工具

- 打开 cydia
- 选择“软件源”->“编辑”->“添加”，输入 APT 地址：



- 添加源完成后，在搜索页面搜索工具并安装。

HBuilder 可以开发 phonegap 应用吗？

首先 HBuilder 的语法提示库是开放的，基于 xml 语法库文件或 jsdoc 这 2 种方式，可以制作任意语法提示。

这部分的开源地址是：<https://github.com/dclaudio/WebFrameworkGrammar>

只要有人制作了 phonegap 语法提示文件，导入到 HBuilder 里就可以提示。

HBuilder 支持 eclipse 插件，开发者完全可开发一个 eclipse 插件装到 HBuilder 上，实现 phonegap 打包等功能。