目录

苹果相关开发总结

- 最新版本： `v1.0`
- 更新时间： `20200620`

鸣谢

感谢我的老婆**陈雪**的包容理解和悉心照料，才使得我 `crifan` 有更多精力去专注技术专研和整理归纳出这些电子书和技术教程，特此鸣谢。

简介

总结Apple苹果的Mac、iOS等平台的开发经验和心得，包括常见的开发工具

源码+浏览+下载

本书的各种源码、在线浏览地址、多种格式文件下载如下：

Gitbook源码

- crifan/apple_develop_summary: 苹果相关开发总结

如何使用此Gitbook源码去生成发布为电子书

详见：crifan/gitbook_template: demo how to use crifan gitbook template and demo

在线浏览

- 苹果相关开发总结 book.crifan.com
- 苹果相关开发总结 crifan.github.io

离线下载阅读

- 苹果相关开发总结 PDF
- 苹果相关开发总结 ePub
- 苹果相关开发总结 Mobi

版权说明

此电子书教程的全部内容，如无特别说明，均为本人原创和整理。其中部分内容参考自网络，均已备注了出处。如有发现侵犯您版权，请通过邮箱联系我 `admin 艾特 crifan.com`，我会尽快删除。谢谢合作。

苹果相关开发概述

此处主要整理苹果Apple相关开发资料和开发工具

- 桌面端
  - `Mac = macOS`
- 移动端
  - 系统
    - `iOS`
  - 设备
    - `iPhone`
  - 开发语言
    - `Swift`
    - `Objective-C`

crifan.com，使用署名4.0国际(CC BY 4.0)协议发布 all right reserved,
powered by Gitbook最后更新： 2020-06-02 11:03:26

通用

此处介绍苹果开发涉及到的通用的内容。

桌面端

苹果开发中桌面端，主要指的是：

- 物理设备：Mac电脑
  - 包括 `Mac Pro` 、 `Mac Air` 等
- 操作系统：
  - 最新叫： `macOS`
  - 之前叫： `OS X`

接下来介绍Mac中各种有用的开发工具。

crifan.com，使用署名4.0国际(CC BY 4.0)协议发布 all right reserved，powered by Gitbook最后更新： 2020-06-02 11:43:34

ideviceinstaller

ideviceinstaller：列出已安装的app信息

在Mac中，想要列查看，通过USB连接上的iOS的设备中，已安装的应用信息，比如app名称，app包名 `bundle id` 等，可以用：

```
ideviceinstaller -l
```

举例：

```
□ ideviceinstaller -l
Total: 37 apps
com.suiyi.foodshop1 - 食行生鲜 4911
com.cisco.anyconnect - AnyConnect 4.6.03052
com.smartisan.reader - 锤子阅读 1311
com.baidu.BaiduMobile - 百度 10.5.5.10
com.ishuyin.iShuYin - 爱书音 1.22
com.evernote.iPhone.Evernote - 印象笔记 358974
com.alipay.iphoneclient - 支付宝 10.1.2.091512
com.autonavi.amap - 高德地图 8.3.0.2104
ctrip.com - 携程旅行 8.3.0
com.Qting.QTTour - 蜻蜓FM 8.0.1.4
com.iflytek.iflyinput - 讯飞输入法 7.0.1815.9602
com.360buy.jdmobile - 京东 7.3.6
com.taobao.tmall - 手机天猫 10948419
com.crifan.voicerecorddemo - 飞语录音Demo 1
org.reactjs.native.example.AwesomeProject - AwesomeProject
com.yingwen.xqlv - 中国象棋 1.01.1
com.crifan.WebDriverAgentRunner.xctrunner - WebDriverAgentF
com.tencent.xin - 微信 6.7.4.44
com.cnvcs.xiangqi - 中国象棋 1.5.0
com.netease.cloudmusic - 网易云音乐 876

com.tencent.mqq - QQ 7.2.9.404
。。。
```

system_profiler

system_profiler：获取系统信息

举例：

从输出中解析出：

真机设备ID

```
system_profiler SPUSBDataType
```

注：

此处输出的是很多非常多的全部的信息：

```
~ □ system_profiler SPUSBDataType
2020-04-29 14:50:16.086 system_profiler[46290:995081] SPUSE
2020-04-29 14:50:16.086 system_profiler[46290:995081] SPUSE
2020-04-29 14:50:16.087 system_profiler[46290:995081] SPUSE
2020-04-29 14:50:16.087 system_profiler[46290:995081] SPUSE
2020-04-29 14:50:16.088 system_profiler[46290:995081] SPUSE
2020-04-29 14:50:16.089 system_profiler[46290:995081] SPUSE
USB:

    USB 3.1 Bus:

        Host Controller Driver: AppleIntelCNLUSBXHCI
        PCI Device ID: 0x9ded
        PCI Revision ID: 0x0030
        PCI Vendor ID: 0x8086

          USB2.1 Hub:

            Product ID: 0x0610
            Vendor ID: 0x05e3  (Genesys Logic, Inc.)
            Version: 6.53
            Speed: Up to 480 Mb/sec
            Manufacturer: GenesysLogic
            Location ID: 0x14200000 / 1
            Current Available (mA): 500
            Current Required (mA): 100
            Extra Operating Current (mA): 0

              iPhone:

                Product ID: 0x12a8
                Vendor ID: 0x05ac (Apple Inc.)
                Version: 7.02
                Serial Number: ed94089f3e34d5538065a695bfdf03
                Speed: Up to 480 Mb/sec
                Manufacturer: Apple Inc.
                Location ID: 0x14230000 / 16
                Current Available (mA): 500
                Current Required (mA): 500
                Extra Operating Current (mA): 0
                Sleep current (mA): 500

              USB Composite Device:

                Product ID: 0x0002
                Vendor ID: 0x0603  (Novatek Microelectronics
                Version: 16.12
                Speed: Up to 1.5 Mb/sec
                Manufacturer: SINO WEALTH
                Location ID: 0x14220000 / 8
```

```
                    Current Available (mA): 500
                    Current Required (mA): 100
                    Extra Operating Current (mA): 0


        USB 3.1 Bus:

          Host Controller Driver: AppleUSBXHCITR
          PCI Device ID: 0x15ec
          PCI Revision ID: 0x0006
          PCI Vendor ID: 0x8086
          Bus Number: 0x00


        USB 3.1 Bus:

          Host Controller Driver: AppleUSBXHCITR
          PCI Device ID: 0x15ec
          PCI Revision ID: 0x0006
          PCI Vendor ID: 0x8086
          Bus Number: 0x01

            USB3.1 Hub:

              Product ID: 0x0626
              Vendor ID: 0x05e3  (Genesys Logic, Inc.)
              Version: 6.53
              Speed: Up to 5 Gb/sec
              Manufacturer: GenesysLogic
              Location ID: 0x01100000 / 1
              Current Available (mA): 900
              Current Required (mA): 0
              Extra Operating Current (mA): 0


        iBridge Bus:

          Host Controller Driver: AppleUSBVHCIBCE

            Touch Bar Backlight:

              Product ID: 0x8102
              Vendor ID: 0x05ac (Apple Inc.)
              Version: 2.01
              Serial Number: 0000000000000000
              Manufacturer: Apple Inc.
              Location ID: 0x80700000

            Touch Bar Display:

              Product ID: 0x8302
              Vendor ID: 0x05ac (Apple Inc.)
              Version: 2.01
              Serial Number: 0000000000000000
```

```
        Manufacturer: Apple Inc.
        Location ID: 0x80600000

    Apple Internal Keyboard / Trackpad:

        Product ID: 0x027b
        Vendor ID: 0x05ac (Apple Inc.)
        Version: 9.27
        Serial Number: FM7845603R8J3VXAH+TVZ
        Speed: Up to 480 Mb/sec
        Manufacturer: Apple Inc.
        Location ID: 0x80500000 / 8
        Current Available (mA): 500
        Current Required (mA): 500
        Extra Operating Current (mA): 0
        Built-In: Yes


    Headset:


        Product ID: 0x8103
        Vendor ID: 0x05ac (Apple Inc.)
        Version: 2.04
        Serial Number: 000000000000
        Manufacturer: Apple
        Location ID: 0x80400000


    Ambient Light Sensor:


        Product ID: 0x8262
        Vendor ID: 0x05ac (Apple Inc.)
        Version: 2.01
        Serial Number: 000000000000
        Manufacturer: Apple Inc.
        Location ID: 0x80300000


    FaceTime HD Camera (Built-in):


        Product ID: 0x8514
        Vendor ID: 0x05ac (Apple Inc.)
        Version: 2.01
        Serial Number: CC28493XQ52J3Y324
        Manufacturer: Apple Inc.
        Location ID: 0x80200000
```

```
        Apple T2 Controller:


          Product ID: 0x8233
          Vendor ID: 0x05ac (Apple Inc.)
          Version: 2.01
          Serial Number: 0000000000000000
          Manufacturer: Apple Inc.
          Location ID: 0x80100000
```

屏幕（宽度和高度等）信息

```
□ system_profiler SPDisplaysDataType
Graphics/Displays:

    Intel Iris Plus Graphics 655:

      Chipset Model: Intel Iris Plus Graphics 655
      Type: GPU
      Bus: Built-In
      VRAM (Dynamic, Max): 1536 MB
      Vendor: Intel
      Device ID: 0x3ea5
      Revision ID: 0x0001
      Metal: Supported, feature set macOS GPUFamily2 v1
      Displays:
        Color LCD:
          Display Type: Built-In Retina LCD
          Resolution: 2560 x 1600 Retina
          Framebuffer Depth: 24-Bit Color (ARGB8888)
          Main Display: Yes
          Mirror: Off
          Online: Yes
          Rotation: Supported
          Automatically Adjust Brightness: No
```

详见：

【已解决】Mac中获取iPhone的分辨率宽高等屏幕信息

获取Mac的序列号

```
□ system_profiler SPHardwareDataType | grep Serial
      Serial Number (system): C02Y3N10JHC8
```

详见：

【已解决】Mac中如何获取笔记本的序列号

端口转发

用 `iproxy` 或 `mobiledevice`

- 如果想要用 `localhost` (或 `127.0.0.1`)去访问（已连接到Mac上的iOS设备）
  - 需要做端口转发
    - `iproxy`
      - 安装：`brew install --HEAD libimobiledevice`
      - 语法：`iproxy <local port> <remote port> [udid]`
      - 用法举例：
        - `iproxy 8100 8100`
          - 当前只连接一个iOS设备时，可以不指定，忽略 `UDID`
        - `iproxy 8100 8100 ed94089f3e34d5538065a695bfdf03dfbb3c5579`
          - 指定对应设备的UDID
            - `ed94089f3e34d5538065a695bfdf03dfb b3c5579` 是此处的iPhone的UDID
          - 可以通过 `idevice_id` 得到
            - `CUR_UDID=$(idevice_id -l | head -n1)`
    - `mobiledevice`
      - 安装：`brew install mobiledevice`
      - 用法：
        - `mobiledevice tunnel 8100 8100`
        - `mobiledevice tunnel -u ed94089f3e34d5538065a695bfdf03dfbb3c5579 8100 8100`
          - 同上，可通过 `-u ios_device_udid`，指定对应iOS设备

iproxy

语法

```
~ □ iproxy --help
usage: iproxy LOCAL_TCP_PORT DEVICE_TCP_PORT [UDID]
```

mobiledevice

语法：

```
~ ▢ mobiledevice help
mobiledevice help
  Display this help screen


mobiledevice version [options]
  Display program version.
  Options:
    -r: Include revision identifier


mobiledevice list_devices [options]
  Display UDID of each connected devices.
  Options:
    -t <timeout>: Timeout (in ms) to wait for devices (defa
    -n <count> : Limit the number of devices to be printed


mobiledevice list_device_props [options]
  List all property names of device.
  Options:
    -u <udid> : Filter by device UDID (default: first detec
    -t <timeout>: Timeout (in ms) to wait for devices (defa


mobiledevice get_device_prop [options] <prop_name>
  Display value of device property with given name.
  Options:
    -u <udid> : Filter by device UDID (default: first detec
    -t <timeout>: Timeout (in ms) to wait for devices (defa


mobiledevice list_apps [options]
  Lists all apps installed on device
  Options:
    -u <udid> : Filter by device UDID (default: first detec
    -t <timeout>: Timeout (in ms) to wait for devices (defa


mobiledevice list_app_props [options] <bundle_id>
  List all property names of app with given bundle id.
  Options:
    -u <udid> : Filter by device UDID (default: first detec
    -t <timeout>: Timeout (in ms) to wait for devices (defa


mobiledevice get_app_prop [options] <bundle_id> <prop_name>
  Display value of app property with given name.
  Options:
    -u <udid> : Filter by device UDID (default: first detec
```

```
              -t <timeout>: Timeout (in ms) to wait for devices (defa


  mobiledevice install_app [options] <path_to_app>
    Install app (.app folder) to device
    Options:
      -u <udid> : Filter by device UDID (default: first detec
      -t <timeout>: Timeout (in ms) to wait for devices (defa



  mobiledevice uninstall_app [options] <bundle_id>
    Uninstall app with given bundle id from device
    Options:
      -u <udid> : Filter by device UDID (default: first detec
      -t <timeout>: Timeout (in ms) to wait for devices (defa



  mobiledevice tunnel [options] <from_port> <to_port>
    Forward TCP connections to connected device
    Options:
      -u <udid> : Filter by device UDID (default: first detec
      -t <timeout>: Timeout (in ms) to wait for devices (defa



  mobiledevice get_bundle_id <path_to_app>
    Display bundle identifier of app (.app folder)
```

instruments

可以用 `instruments` 列出当前所有苹果的设备

举例：

```
~ □ instruments -s
CoreData: annotation:  Failed to load optimized model at pa
Known Devices:
limao的MacBook Pro [F9089371-1060-5CE3-99BB-81741693BE80]
Crifan iPhone6 (12.4.5) [ed94089f3e34d5538065a695bfdf03dfbb
Apple TV (13.3) [6680F059-4DE1-430C-B696-228AC27CAA88] (Sim
Apple TV 4K (13.3) [048E58E8-6A27-4D81-BDEB-8812C610B756] (
Apple TV 4K (at 1080p) (13.3) [384D5E60-B6B1-481E-BDC3-B7FF
Apple Watch Series 4 - 40mm (6.1.1) [1B98415B-3FDE-401B-A80
Apple Watch Series 4 - 44mm (6.1.1) [661838E9-B0BE-42B4-B55
iPad (7th generation) (13.3) [7F8EDE89-74E0-4BAB-B3CA-09E2D
iPad Air (3rd generation) (13.3) [BBC48526-3922-4C97-BA14-B
iPad Pro (11-inch) (13.3) [04DD3B8A-5B78-48E8-8B22-56796A9C
iPad Pro (12.9-inch) (3rd generation) (13.3) [D811684E-2F3B
iPad Pro (9.7-inch) (13.3) [B11D5D40-FEA2-4114-B053-E4CFD29
iPhone 11 (13.3) [509BC7C7-9C0E-42FA-8AB2-F5220EBAA13B] (Si
iPhone 11 Pro (13.3) [3E8E7E92-66F2-4AF3-A405-23B5FB231DE7]
iPhone 11 Pro (13.3) + Apple Watch Series 5 - 40mm (6.1.1)
iPhone 11 Pro Max (13.3) [50C15135-1532-44C5-B82C-B327F88F2
iPhone 11 Pro Max (13.3) + Apple Watch Series 5 - 44mm (6.1
iPhone 8 (13.3) [54589698-0C9F-407D-B21A-83432CABB681] (Sim
iPhone 8 Plus (13.3) [509B7103-97DB-4AB9-B829-001190ED4B7E]
Known Templates:
"Activity Monitor"
"Allocations"
"App Launch"
"Blank"
"Core Animation"
"Core Data"
"Counters"
"Energy Log"
"File Activity"
"Game Performance"
"Leaks"
"Metal System Trace"

"Network"
"SceneKit"
"SwiftUI"
"System Trace"
"Time Profiler"
"Zombies"
```

idevice_id

可以用 `idevice_id` 列出当前（Mac中已）连接的（iOS）的设备

举例：

```
▢ idevice_id -l
ed94089f3e34d5538065a695bfdf03dfbb3c5579
```

帮助和语法

```
~ ▢ idevice_id --help
Usage: idevice_id [OPTIONS] [UDID]
Prints device name or a list of attached devices.

  If UDID is given, the name of the connected device with

  -l, --list      list UDIDs of all devices attached via US
  -n, --network   list UDIDs of all devices available via r
  -d, --debug     enable communication debugging
  -h, --help      prints usage information

Homepage: <http://libimobiledevice.org>
```

移动端

iOS

iOS自动化

详见完整教程：

移动端自动化测试概览

中的

iOS自动化测试利器：facebook-wda

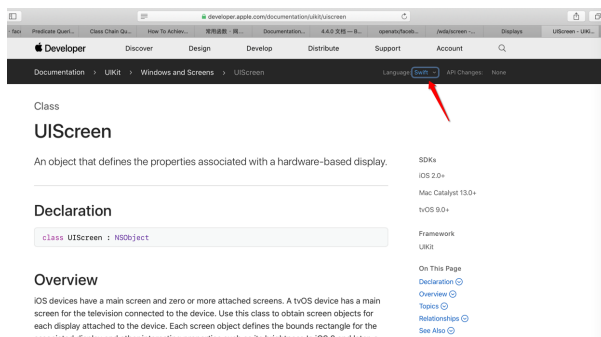# 附录

下面列出相关参考资料。

文档

此处整理出有用的苹果相关开发文档，供需要时查阅。

iOS 旧文档

- 举例
    - Displays - iOS Device Compatibility Reference

iOS 新文档

统一后的 = 新的

- 举例
    - UIScreen - UIKit | Apple Developer Documentation
- 且还分2种语言
    - `Swift`
        - https://developer.apple.com/documentation/uikit/uiscreen
            - 截图

    - `Objective-C`
        - https://developer.apple.com/documentation/uikit/uiscreen?language=objc
            - 截图
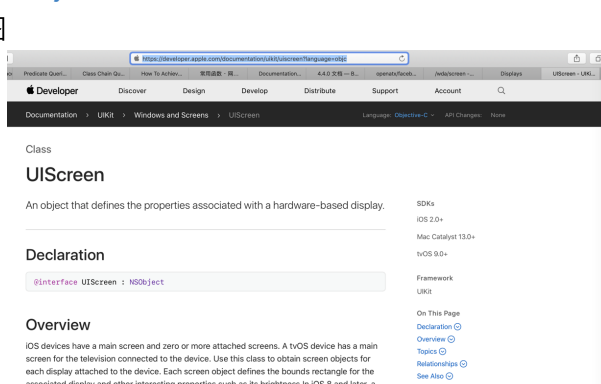
- 说明
    - 其中内容大体类似，主要是语法不同

- 不过有时候细节也不太一样
- 举例
  - Objective-C 中 `UIDeviceBatteryState` 中，找不到枚举值的常量值定义

[UIDeviceBatteryState - UIKit | Apple Developer Documentation]()

```
typedef enum UIDeviceBatteryState : NSInteger {
    ...
} UIDeviceBatteryState;
```

- Constants
  - UIDeviceBatteryStateUnknown
    - The battery state for the device cannot be determined.
  - UIDeviceBatteryStateUnplugged
    - The device is not plugged into power; the battery is discharging.
  - UIDeviceBatteryStateCharging
    - The device is plugged into power and the battery is less than 100% charged.
  - UIDeviceBatteryStateFull
    - The device is plugged into power and the battery is 100% charged.

-》想要知道对应的枚举值定义的int值

-》

[UIDeviceBatteryStateUnknown - UIDeviceBatteryState | Apple Developer Documentation]()

> The battery state for the device cannot be determined.
>
> UIDeviceBatteryStateUnknown

-》没看到定义的值

-》无意间发现，换Swift语言后：

[UIDevice.BatteryState - UIDevice | Apple Developer Documentation]()

-》点击具体的某个定义，都可以看到具体常量枚举值定义：

- case unknown
  - The battery state for the device cannot be determined.
    - UIDevice.BatteryState.unknown - UIDevice.BatteryState | Apple Developer Documentation
    - https://developer.apple.com/documentation/uikit/uidevice/batterystate/unknown

- case unknown = 0
- case unplugged
  - The device is not plugged into power; the battery is discharging.
    - UIDevice.BatteryState.unplugged - UIDevice.BatteryState | Apple Developer Documentation
    - https://developer.apple.com/documentation/uikit/uidevice/batterystate/unplugged
    - case unplugged = 1
- case charging
  - The device is plugged into power and the battery is less than 100% charged.
    - UIDevice.BatteryState.charging - UIDevice.BatteryState | Apple Developer Documentation
    - https://developer.apple.com/documentation/uikit/uidevice/batterystate/charging
    - case charging = 2
- case full
  - The device is plugged into power and the battery is 100% charged.
    - UIDevice.BatteryState.full - UIDevice.BatteryState | Apple Developer Documentation
    - https://developer.apple.com/documentation/uikit/uidevice/batterystate/full
    - case full = 3

-》总结来说：

此处是 `ObjC` 中看不到 `UIDeviceBatteryState` 的细节的枚举值的定义

而换成 `Swift` 的 `UIDevice.BatteryState`，就可以看到具体的枚举的定义的常量的值了

参考资料

- 【已解决】Mac中获取iPhone的分辨率宽高等屏幕信息
- 【已解决】Mac中如何获取笔记本的序列号
-
- 移动端自动化测试概览
- iOS自动化测试利器：facebook-wda
-
- 使用自定义 WDA 服务器 - Appium
- 使用 Python 库 facebook-wda 完成网易云音乐 iOS 客户端的自动化测试 (示例) · TesterHome
- Displays - iOS Device Compatibility Reference
- UIScreen - UIKit | Apple Developer Documentation
- UIDeviceBatteryState - UIKit | Apple Developer Documentation
- UIDeviceBatteryStateUnknown - UIDeviceBatteryState | Apple Developer Documentation
- UIDevice.BatteryState - UIDevice | Apple Developer Documentation
- iOS真机安装WebDriverAgent | Vicの博客
-