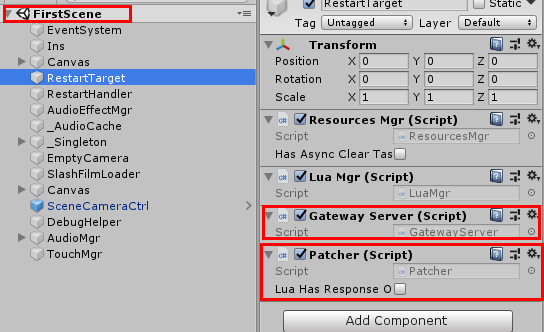
# 更新

## 运行流程

### FirstScene



## 打包流程

### 版本流程

MainVersion.SubVersion.PatchVersion

主板本号.子版本号.补丁版本号

其中只有补丁版本才是热更新 其余都要强更

大版本= 主板本号.子版本号

大版本= MainVersion.SubVersion

客户端流程

1.拉取MAX\_VERSION.txt服务器最新版本

2.对比版本信息确认是否需要热更或者强更

3.需要版本热更的话 拉取FULL\_VERSIONS.json 做进一步检查具体版本 IOS 是FULL\_VERSION\_IOS.json

4.下载地址会是cdn的patch目录下的android\_1.8.1.zip 或者ios\_1.8.1.zip

补丁制作工具

1.补丁工具是运行在Unity环境下的工具 所有目录(未特殊说明)都是在Unity的根目录下

2.包含的功能有：

1.制作ab包

2.对比svn版本,找出diff文件

3.生成zip文件和更新FULL\_VERSIONS.json文件

4.上传到FTP cdn服务器 并且反向校验文件md5

3.制作的补丁等信息都在VersionFiles目录下

4.上传信息配置是在ftp\_config.txt 第一行目标ftp地址 第二行账户 第三行密码

补丁制作流程

1.方法1：(一键式操作从ab生成svn提取zip制作等)

1.利用菜单 补丁工具/------一键生成IOS补丁信息 VERSION.txt ，

会自动工具这txt的版本号进行生成并且最终制作好zip文件和更新json文件

2.利用菜单 补丁工具/FTP上传Android补丁包 VERSION.txt ，

会自动工具txt的版本上传zip到FTP

2.方法2：(其实是把方法1细化,可以把4步独立执行)

1.

大版本制作流程

1.菜单 AssetBundle/BuildAssetBundle 会吧当前unity的平台的ab包输出到目录AssetBundles

2.copy 目录到 Assets/StreamingAssets

3.unity直接build即可，注意要更改DevConfig.cs里面的版本和unity打包的版本号对应上

4.更新服务器MAX\_VERSION.txt 为版本号即可

5.注意大版本是可以携带patch发布的 比如大版本1.9.5

6.一个apk包是可以多次发布 是为了方式下载的apk包 需要更新的patch太多 制作方法和大版本一样 只不过是兼容的

补丁工具实现具体说明(有兴趣的话可以看看)

1.制作ab包

1.工具约定的RawResources目录级别结构输出ab到AssetBundles

2.检查svn 找出diff

1.把AssetBundles 目录copy到完整的svn信息目录(VersionFiles\AssetBundles\Android\)

然后基于此目录找出diff文件并且copy到同级的AssetBundles\_diff

2.目录AssetBundles\_diff 即是打包zip的目录 这是一个临时目录每次会重新生成不需要上传SVN

3.因此copy到的目录会始终是一个版本 每个版本出后 都要同步到SVN以便下次

补丁的diff处理

3.生成zip

1.把AssetBundles\_diff目录 根据VERSION.txt生成对应版本的zip文件例如 VersionFiles\1.8.1\adnroid\_1.8.1.zip

2.更新同大版本的补丁json信息到FULL\_VERSIONS.json 该文件只记录了当前大版本下的所有补丁信息 TODO 这个可以文件大小优化一下

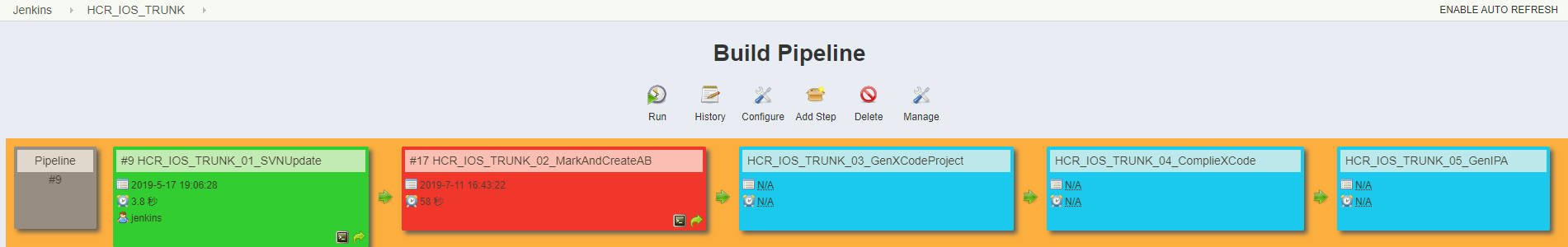
# 工具

StickNote

VNote

# Jenkins

## HCR\_IOS\_TRUNK



### HCR\_IOS\_TRUNK\_01\_SVNUpdate

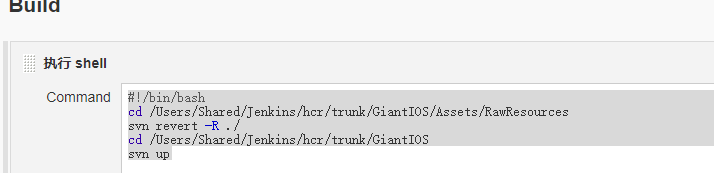
#!/bin/bash

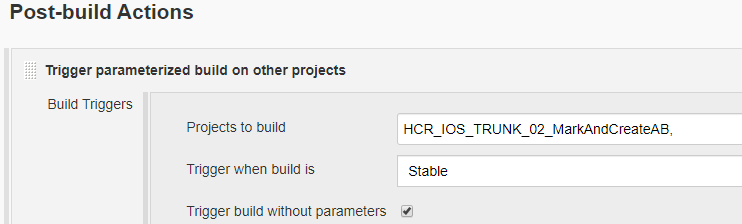
cd /Users/Shared/Jenkins/hcr/trunk/GiantIOS/Assets/RawResources

svn revert -R ./

cd /Users/Shared/Jenkins/hcr/trunk/GiantIOS

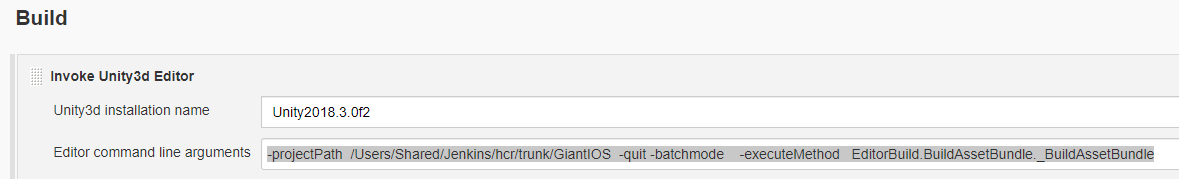
svn up

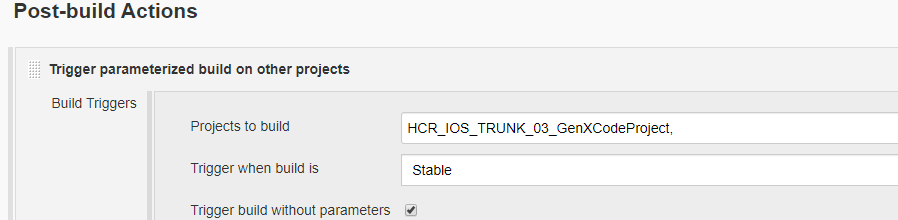




### HCR\_IOS\_TRUNK\_02\_MarkAndCreateAB

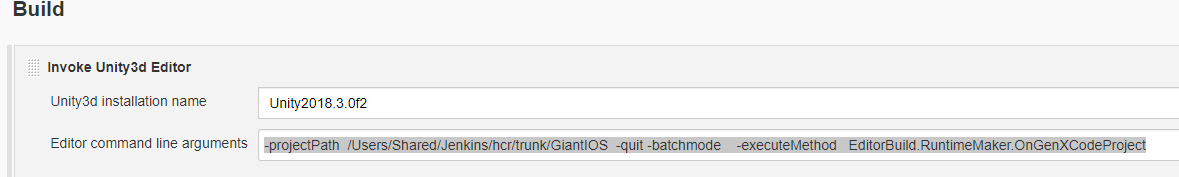
-projectPath /Users/Shared/Jenkins/hcr/trunk/GiantIOS -quit -batchmode -executeMethod EditorBuild.BuildAssetBundle.\_BuildAssetBundle

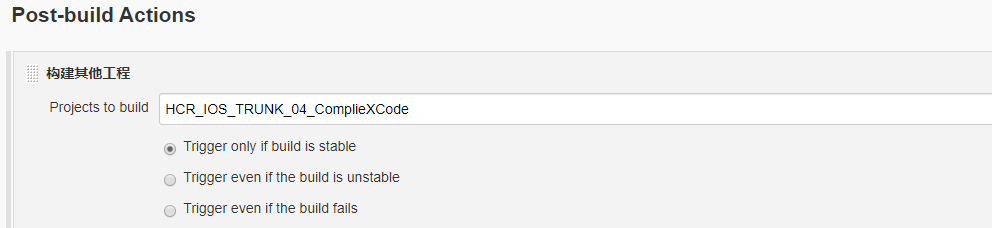




### HCR\_IOS\_TRUNK\_03\_GenXCodeProject

-projectPath /Users/Shared/Jenkins/hcr/trunk/GiantIOS -quit -batchmode -executeMethod EditorBuild.RuntimeMaker.OnGenXCodeProject





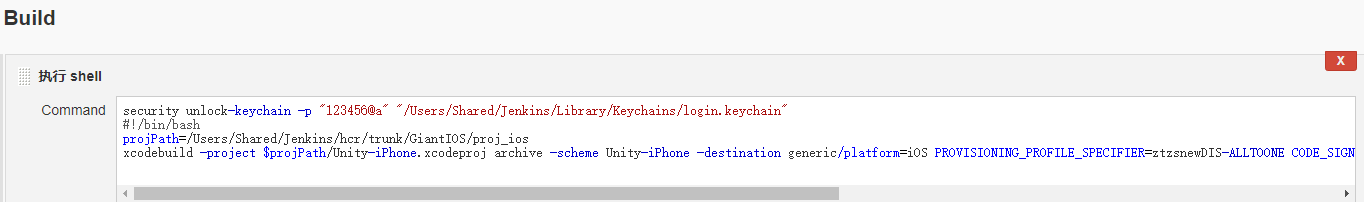
### HCR\_IOS\_TRUNK\_04\_ComplieXCode

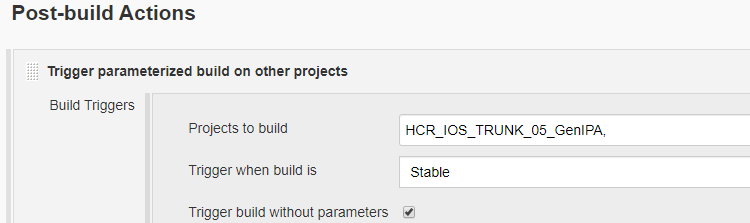
security unlock-keychain -p "123456@a" "/Users/Shared/Jenkins/Library/Keychains/login.keychain"

#!/bin/bash

projPath=/Users/Shared/Jenkins/hcr/trunk/GiantIOS/proj\_ios

xcodebuild -project $projPath/Unity-iPhone.xcodeproj archive -scheme Unity-iPhone -destination generic/platform=iOS PROVISIONING\_PROFILE\_SPECIFIER=ztzsnewDIS-ALLTOONE CODE\_SIGN\_IDENTITY="iPhone Distribution: Shanghai Giant Network Technology Co., Ltd." -archivePath $projPath/Unity-iPhone.xcarchive





### HCR\_IOS\_TRUNK\_05\_GenIPA

#!/bin/bash

projPath=/Users/Shared/Jenkins/hcr/trunk/GiantIOS/proj\_ios

xcodebuild -exportArchive -archivePath $projPath/Unity-iPhone.xcarchive -exportPath $projPath/out -exportOptionsPlist $projPath/../build.plist

cp $projPath/out/Unity-iPhone.ipa /Library/WebServer/Documents/jdhcr/ios/jdhcr-$(date +%m%d%H%M%y).ipa

mv $projPath/out/Unity-iPhone.ipa /Library/WebServer/Documents/jdhcr/ios/jdhcr2019.ipa



## ios\_branch\_18.6.0

### ios\_branche\_01\_svnupdate

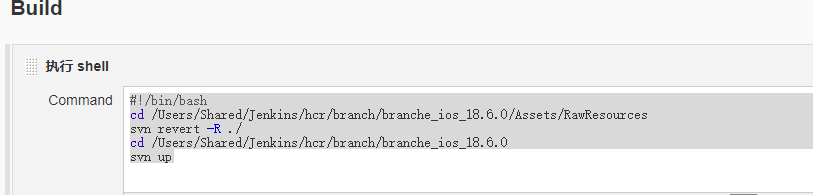
#!/bin/bash

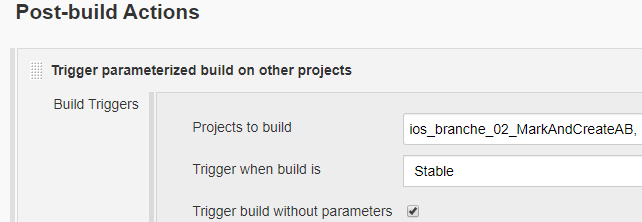
cd /Users/Shared/Jenkins/hcr/branch/branche\_ios\_18.6.0/Assets/RawResources

svn revert -R ./

cd /Users/Shared/Jenkins/hcr/branch/branche\_ios\_18.6.0

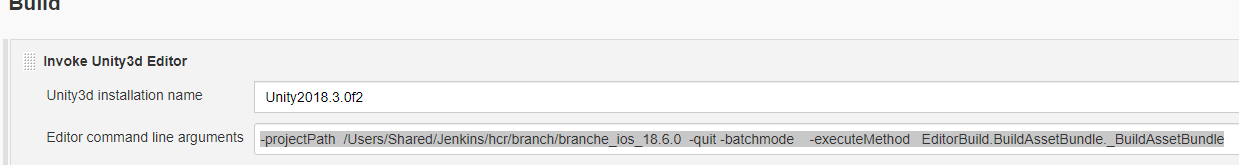
svn up

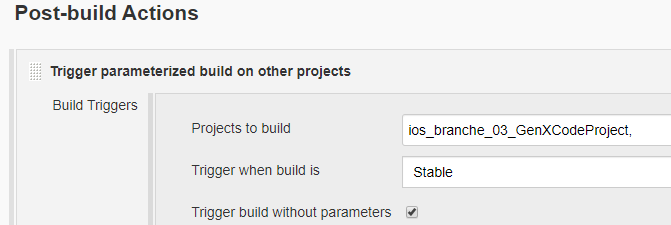




### ios\_branche\_02\_MarkAndCreateAB

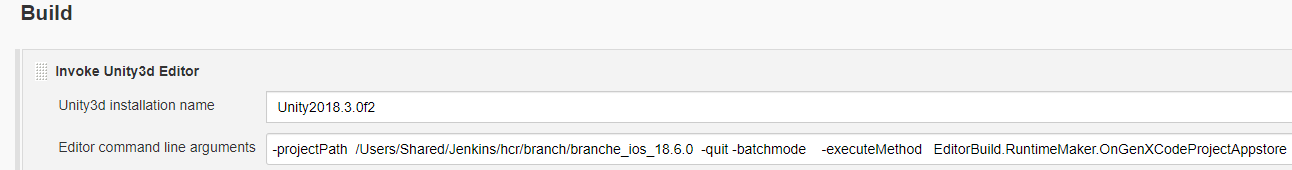
-projectPath /Users/Shared/Jenkins/hcr/branch/branche\_ios\_18.6.0 -quit -batchmode -executeMethod EditorBuild.BuildAssetBundle.\_BuildAssetBundle

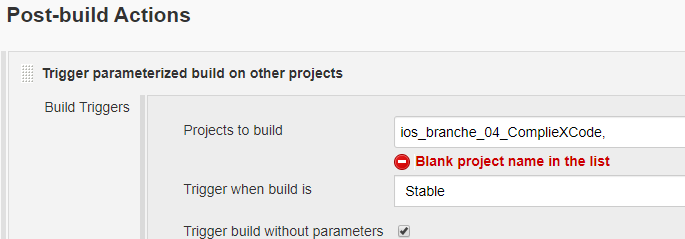




### ios\_branche\_03\_GenXCodeProject

-projectPath /Users/Shared/Jenkins/hcr/branch/branche\_ios\_18.6.0 -quit -batchmode -executeMethod EditorBuild.RuntimeMaker.OnGenXCodeProjectAppstore





### ios\_branche\_04\_ComplieXCode

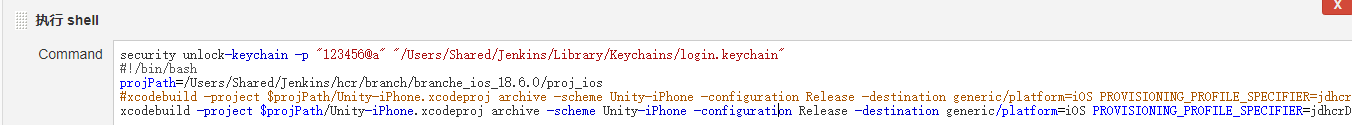
security unlock-keychain -p "123456@a" "/Users/Shared/Jenkins/Library/Keychains/login.keychain"

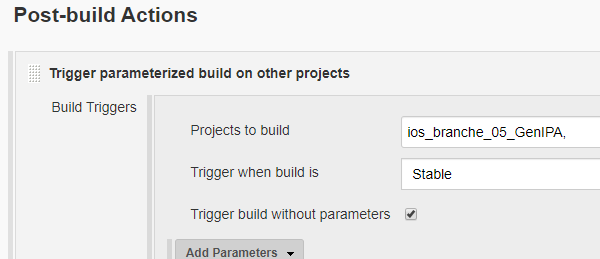
#!/bin/bash

projPath=/Users/Shared/Jenkins/hcr/branch/branche\_ios\_18.6.0/proj\_ios

#xcodebuild -project $projPath/Unity-iPhone.xcodeproj archive -scheme Unity-iPhone -configuration Release -destination generic/platform=iOS PROVISIONING\_PROFILE\_SPECIFIER=jdhcrDIS CODE\_SIGN\_IDENTITY="iPhone Distribution: Shanghai Zhengju Information Technology Co., Ltd. (TG93K6XG5H)" -archivePath $projPath/Unity-iPhone.xcarchive

xcodebuild -project $projPath/Unity-iPhone.xcodeproj archive -scheme Unity-iPhone -configuration Release -destination generic/platform=iOS PROVISIONING\_PROFILE\_SPECIFIER=jdhcrDEV CODE\_SIGN\_IDENTITY="iPhone Developer: junjie zhang (N86KABWGCZ)" -archivePath $projPath/Unity-iPhone.xcarchive





### ios\_branche\_05\_GenIPA

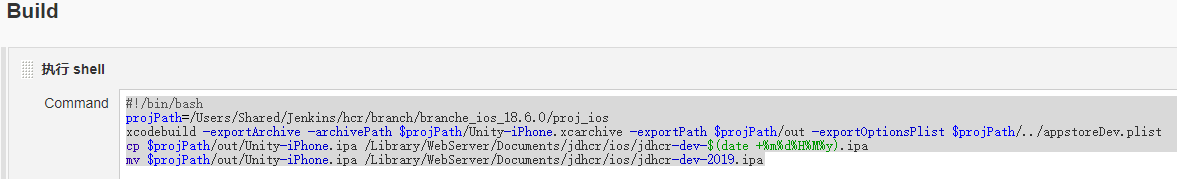
#!/bin/bash

projPath=/Users/Shared/Jenkins/hcr/branch/branche\_ios\_18.6.0/proj\_ios

xcodebuild -exportArchive -archivePath $projPath/Unity-iPhone.xcarchive -exportPath $projPath/out -exportOptionsPlist $projPath/../appstoreDev.plist

cp $projPath/out/Unity-iPhone.ipa /Library/WebServer/Documents/jdhcr/ios/jdhcr-dev-$(date +%m%d%H%M%y).ipa

mv $projPath/out/Unity-iPhone.ipa /Library/WebServer/Documents/jdhcr/ios/jdhcr-dev-2019.ipa



## ios\_tmp\_dis

### ios\_branche\_04\_CompileXCode\_Appstore-dis

security unlock-keychain -p "123456@a" "/Users/Shared/Jenkins/Library/Keychains/login.keychain"

#!/bin/bash

projPath=/Users/Shared/Jenkins/hcr/branch/branche\_ios\_18.6.0/proj\_ios

#disappstoreprofile

#xcodebuild -project $projPath/Unity-iPhone.xcodeproj archive -scheme Unity-iPhone -configuration Release -destination generic/platform=iOS PROVISIONING\_PROFILE\_SPECIFIER=jdhcrDIS CODE\_SIGN\_IDENTITY="iPhone Distribution: Shanghai Zhengju Information Technology Co., Ltd. (TG93K6XG5H)" -archivePath $projPath/Unity-iPhone.xcarchive

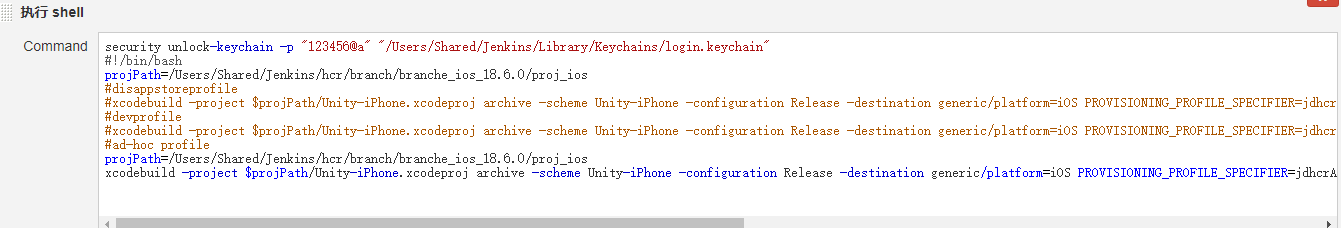
#devprofile

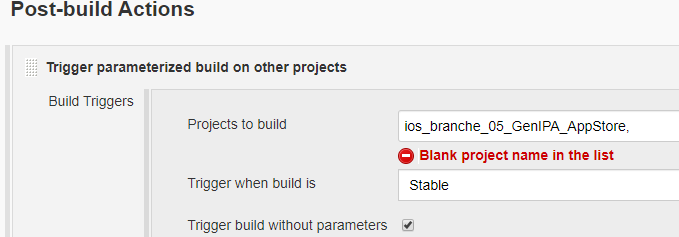
#xcodebuild -project $projPath/Unity-iPhone.xcodeproj archive -scheme Unity-iPhone -configuration Release -destination generic/platform=iOS PROVISIONING\_PROFILE\_SPECIFIER=jdhcrDEV CODE\_SIGN\_IDENTITY="iPhone Developer: junjie zhang (N86KABWGCZ)" -archivePath $projPath/Unity-iPhone.xcarchive

#ad-hoc profile

projPath=/Users/Shared/Jenkins/hcr/branch/branche\_ios\_18.6.0/proj\_ios

xcodebuild -project $projPath/Unity-iPhone.xcodeproj archive -scheme Unity-iPhone -configuration Release -destination generic/platform=iOS PROVISIONING\_PROFILE\_SPECIFIER=jdhcrADHOCdis CODE\_SIGN\_IDENTITY="iPhone Distribution: Shanghai Zhengju Information Technology Co., Ltd. (TG93K6XG5H)" -archivePath $projPath/Unity-iPhone.xcarchive





### ios\_branche\_05\_GenIPA\_AppStore

#!/bin/bash

projPath=/Users/Shared/Jenkins/hcr/branch/branche\_ios\_18.6.0/proj\_ios

xcodebuild -exportArchive -archivePath $projPath/Unity-iPhone.xcarchive -exportPath $projPath/out -exportOptionsPlist $projPath/../appstoreAdHoc.plist

cp $projPath/out/Unity-iPhone.ipa /Library/WebServer/Documents/jdhcr/ios/jdhcr-dis-$(date +%m%d%H%M%y).ipa

mv $projPath/out/Unity-iPhone.ipa /Library/WebServer/Documents/jdhcr/ios/jdhcr-dis-2019.ipa



## zttest(all green)

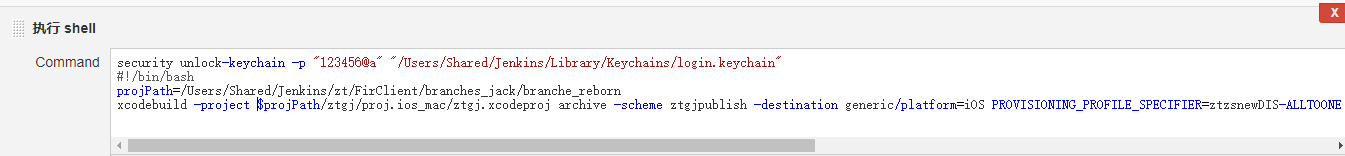
### ztapp\_ios\_002\_complie

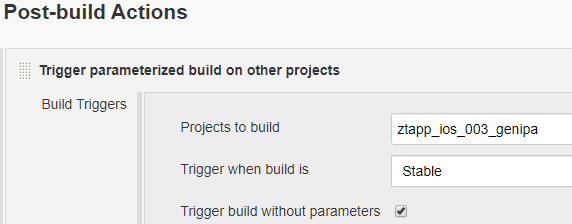
security unlock-keychain -p "123456@a" "/Users/Shared/Jenkins/Library/Keychains/login.keychain"

#!/bin/bash

projPath=/Users/Shared/Jenkins/zt/FirClient/branches\_jack/branche\_reborn

xcodebuild -project $projPath/ztgj/proj.ios\_mac/ztgj.xcodeproj archive -scheme ztgjpublish -destination generic/platform=iOS PROVISIONING\_PROFILE\_SPECIFIER=ztzsnewDIS-ALLTOONE CODE\_SIGN\_IDENTITY="iPhone Distribution: Shanghai Giant Network Technology Co., Ltd." -archivePath $projPath/Ztgj-iPhone.xcarchive



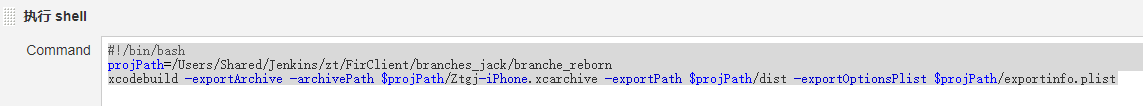


### ztapp\_ios\_003\_genipa

#!/bin/bash

projPath=/Users/Shared/Jenkins/zt/FirClient/branches\_jack/branche\_reborn

xcodebuild -exportArchive -archivePath $projPath/Ztgj-iPhone.xcarchive -exportPath $projPath/dist -exportOptionsPlist $projPath/exportinfo.plist



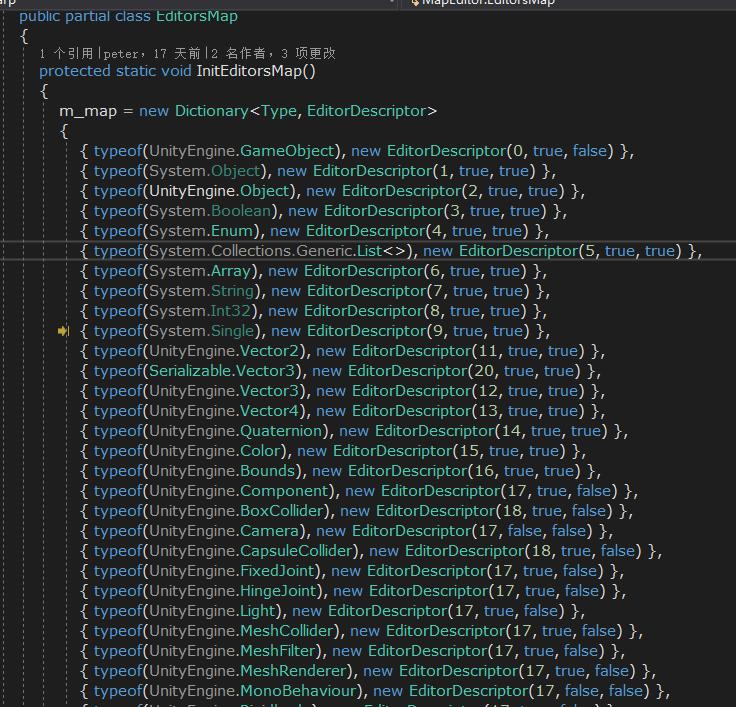
# Macpro

# 地编

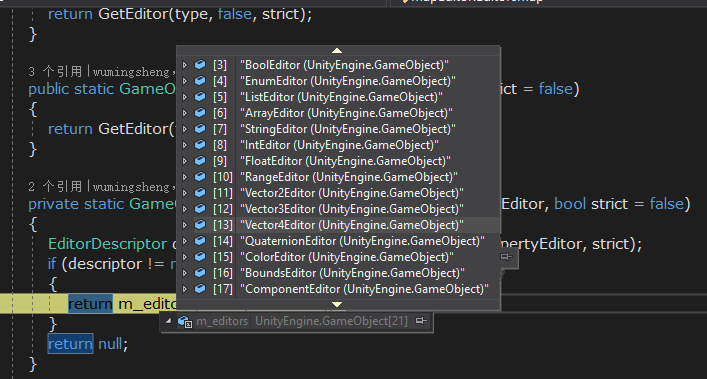
组合（搭积木）

## 属性编辑

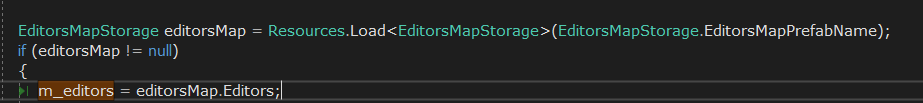
### 类型定义



### 编辑原型：



Resource下的MapEditorsMap保存了所有编辑原型



**ComponentEditor**

**//editor:根据模板新创建出的UI, ComponentsPanel:是gameObjectEditor下的专门用来挂组件UI的**

**editor**.transform.SetParent(ComponentsPanel, false);

**editor**.Component = component;



### 组件描述

# Reflection

## 获取属性：

PropertyInfo prop = type.GetProperty("enabled", BindingFlags.Public | BindingFlags.DeclaredOnly | BindingFlags.Instance);

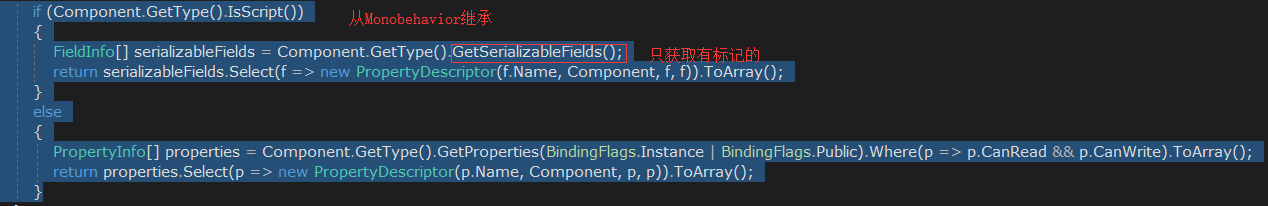
## 属性的类型：

prop.**PropertyType**

## 取得类型的名字：

Component.GetType().Name;

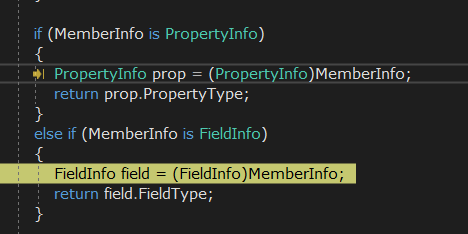
## 取得所有属性



## 判断子类：

type.IsSubclassOf(typeof(MonoBehaviour));

## 判断是PropertyInfo还是FieldInfo



## 根据类型取得所有可以序列号的字段

FieldInfo[] serializableFields = Reflection.**GetSerializableFields**(memberInfo.GetType());

## 根据filed取value

com:实体

field.GetValue(com)

## XXXX

**fi.GetType()**

**和**

**fi.FieldType**

**是两回事**