

# 放心课App包大小分析

## 安装包的构成

首先来看下，线上最新3.2.0版本的包大小

ACFunReleaseV3.2.0B1192.ipa	2022年9月9日 11:15	146.8 MB	iOS 软件包归档
DistributionSummary.plist	2022年9月9日 11:15	66 KB	Property List
ExportOptions.plist	2022年9月9日 11:15	739 字节	Property List
Packaging.log	2022年9月9日 11:15	1.1 MB	日志文件

接入新版IM后，App包大小后**144.9MB**，该包大小是在移除了一些无用依赖后的大小

名称	修改日期	大小	种类
放心课.ipa	今天 14:49	144.9 MB	OS 软件包归档
DistributionSummary.plist	今天 14:50	55 KB	Property List
ExportOptions.plist	今天 14:50	701 字节	Property List
Packaging.log	今天 14:49	961 KB	日志文件

通过解压ipa包，可以看到，其中Payload文件夹包含放心课.app，SwiftSupport文件夹稍后介绍

> Payload	今天 14:49	-- 文件夹
> SwiftSupport	今天 14:48	-- 文件夹

放心课.app结构如下

```

→ ~ /Users/admin/Desktop/ACFun\ 2022-10-18\ 14-50-24/放心课的副本/Payload/放心课.app/
→ 放心课.app du -shc * | sort -rh
188M    total
178M    Frameworks
4.4M    放心课
3.9M    Assets.car
496K    LaunchScreen1@3x.png
244K    LaunchScreen1@2x.png
140K    _CodeSignature
92K     vconsole.js
68K     GTCaptcha4.bundle
24K     Base.lproj
12K     embedded.mobileprovision
8.0K    AppIcon76x76@2x~ipad.png
8.0K    AppIcon60x60@2x.png
4.0K    dealMCallCompatible.js
4.0K    PkgInfo
4.0K    Info.plist
4.0K    DripPay.bundle

```

其中Frameworks文件夹占用了**178MB**，由于我们这边依赖的第三方库基本都是动态库，所以依赖的外部库基本都在这个文件夹下

**说明：Frameworks文件夹里面包含嵌入的swift语言动态库和第三方依赖库**

使用 `otool -L` 查看了依赖的动态库，发现包含了两种指令集**armv7**和**arm64**

```

/Users/admin/Desktop/ACFun\ 2022-10-18\ 14-50-24/放心课的副本/Payload/放心课.app/Frameworks/AFNetworking.framework
→ AFNetworking.framework otool -L AFNetworking
AFNetworking (architecture armv7):
 @rpath/AFNetworking.framework/AFNetworking (compatibility version 1.0.0, current version 1.0.0)
 /System/Library/Frameworks/Foundation.framework/Foundation (compatibility version 300.0.0, current version 1775.118.101)
 /usr/lib/libobjc.A.dylib (compatibility version 1.0.0, current version 228.0.0)
 /usr/lib/libSystem.B.dylib (compatibility version 1.0.0, current version 1292.100.5)
 /System/Library/Frameworks/CoreFoundation.framework/CoreFoundation (compatibility version 150.0.0, current version 1775.118.101)
 /System/Library/Frameworks/CoreGraphics.framework/CoreGraphics (compatibility version 64.0.0, current version 1463.4.1)
 /System/Library/Frameworks/MobileCoreServices.framework/MobileCoreServices (compatibility version 1.0.0, current version 1122.33.0)
 /System/Library/Frameworks/Security.framework/Security (compatibility version 1.0.0, current version 59754.100.106)
 /System/Library/Frameworks/SystemConfiguration.framework/SystemConfiguration (compatibility version 1.0.0, current version 1109.100.4)
 /System/Library/Frameworks/UIKit.framework/UIKit (compatibility version 1.0.0, current version 4218.1.100)
 /System/Library/Frameworks/WebKit.framework/WebKit (compatibility version 1.0.0, current version 611.1.21)
AFNetworking (architecture arm64):
 @rpath/AFNetworking.framework/AFNetworking (compatibility version 1.0.0, current version 1.0.0)
 /System/Library/Frameworks/Foundation.framework/Foundation (compatibility version 300.0.0, current version 1775.118.101)
 /usr/lib/libobjc.A.dylib (compatibility version 1.0.0, current version 228.0.0)
 /usr/lib/libSystem.B.dylib (compatibility version 1.0.0, current version 1292.100.5)
 /System/Library/Frameworks/CoreFoundation.framework/CoreFoundation (compatibility version 150.0.0, current version 1775.118.101)
 /System/Library/Frameworks/CoreGraphics.framework/CoreGraphics (compatibility version 64.0.0, current version 1463.4.1)
 /System/Library/Frameworks/MobileCoreServices.framework/MobileCoreServices (compatibility version 1.0.0, current version 1122.33.0)
 /System/Library/Frameworks/Security.framework/Security (compatibility version 1.0.0, current version 59754.100.106)
 /System/Library/Frameworks/SystemConfiguration.framework/SystemConfiguration (compatibility version 1.0.0, current version 1109.100.4)
 /System/Library/Frameworks/UIKit.framework/UIKit (compatibility version 1.0.0, current version 4218.1.100)
 /System/Library/Frameworks/WebKit.framework/WebKit (compatibility version 1.0.0, current version 611.1.21)

```

其中**armv7**包含机型如下

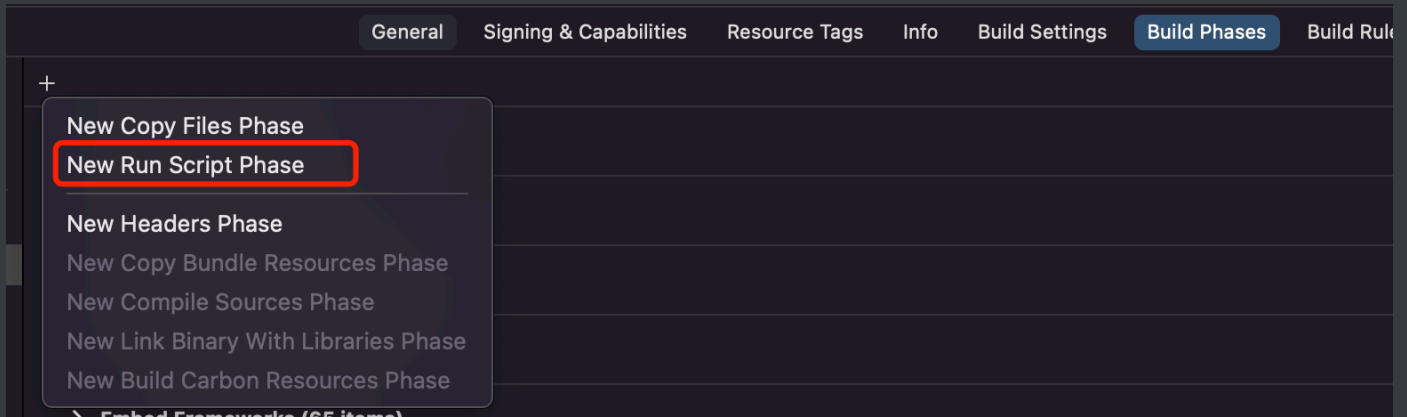
```

armv7: iPhone4 | iPhone4S | iPad | iPad2 | iPad3 | iPad mini | iPod Touch 3G |
iPod Touch4

```

由于这些机型已经不再被支持（我们最低支持的系统版本为*iOS 11*），所有可以把\*\*armv7指令集给移除

移除该指令集方式很简单，只需要选中**Build Phases**，点击+号，选中**New Run Script Phase**



配置脚本如下：

```
APP_PATH="${TARGET_BUILD_DIR}/${WRAPPER_NAME}"

# This script loops through the frameworks embedded in the application
and
# removes unused architectures.
find "$APP_PATH" -name '*.framework' -type d | while read -r FRAMEWORK
do
FRAMEWORK_EXECUTABLE_NAME=$(defaults read "$FRAMEWORK/Info.plist"
CFBundleExecutable)
FRAMEWORK_EXECUTABLE_PATH="$FRAMEWORK/$FRAMEWORK_EXECUTABLE_NAME"
echo "Executable is $FRAMEWORK_EXECUTABLE_PATH"

NON_FAT_FILE="$(lipo -info "$FRAMEWORK_EXECUTABLE_PATH" | cut -d ':' -
f1)"

if [[ "$NON_FAT_FILE" != "Architectures in the fat file" ]]; then
    continue
fi
```

```

EXTRACTED_ARCHS=()

for ARCH in $ARCHS
do
echo "Extracting $ARCH from $FRAMEWORK_EXECUTABLE_NAME"
lipo -extract "$ARCH" "$FRAMEWORK_EXECUTABLE_PATH" -o
"$FRAMEWORK_EXECUTABLE_PATH-$ARCH"
EXTRACTED_ARCHS+=("$FRAMEWORK_EXECUTABLE_PATH-$ARCH")
done

echo "Merging extracted architectures: ${ARCHS}"
lipo -o "$FRAMEWORK_EXECUTABLE_PATH-merged" -create
"${EXTRACTED_ARCHS[@]}"
rm "${EXTRACTED_ARCHS[@]}"

echo "Replacing original executable with thinned version"
rm "$FRAMEWORK_EXECUTABLE_PATH"
mv "$FRAMEWORK_EXECUTABLE_PATH-merged" "$FRAMEWORK_EXECUTABLE_PATH"

done

```

移除后armv7指令集后，ipa包大小为124.3MB，大概较少了20MB，还是很可观的

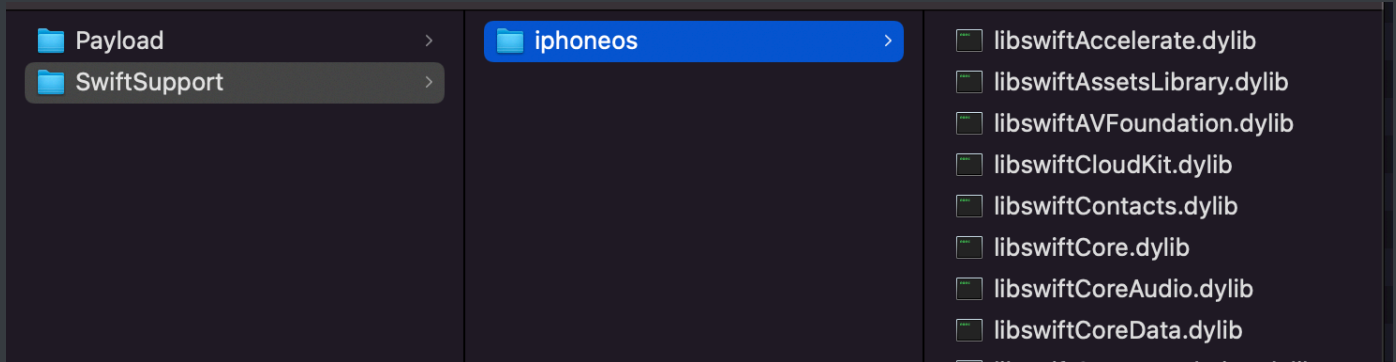
名称	修改日期	大小	种类
放心课.ipa	今天 15:24	124.3 MB	iOS 软件包归档
DistributionSummary.plist	今天 15:24	52 KB	Property List
ExportOptions.plist	今天 15:24	701 字节	Property List
Packaging.log	今天 15:24	951 KB	日志文件

## SwiftSupport处理

前面我们在解压ipa包后，发现有个SwiftSupport文件夹，占了158MB

```
→ ~ /Users/admin/Desktop/ACFun\ 2022-10-18\ 15-24-40/放心课
→ 放心课 du -shc * | sort -rh
289M    total
158M    SwiftSupport
131M    Payload
```

结构如下：



发现里面都是一些 swift 语言动态库

由于这些库都是 swift 语言动态库，应该会被包含在iOS系统里面，不会单独给打进每个ipa包里面，带着这个疑问，就去App Store上下载了我们放心课App包，通过解压发现，文件夹结构如下：

```
→ 放心课 3.2.0 l
total 96
drwx-----@ 6 admin  staff  192B  10  12  16:39  .
drwx-----@ 55 admin  staff  1.7K  10  12  17:50  ..
drwxr-xr-x@ 4 admin  staff  128B  9   9  11:41  META-INF
drwxr-xr-x@ 4 admin  staff  128B  10  12  17:13  Payload
-rw-r--r--@ 1 admin  staff  43K   10  12  16:38  iTunesArtwork
-rw-r--r--@ 1 admin  staff  3.8K  10  12  16:38  iTunesMetadata.plist
→ 放心课 3.2.0 cd Payload
→ Payload l
total 16
drwxr-xr-x@ 4 admin  staff  128B  10  12  17:13  .
drwx-----@ 6 admin  staff  192B  10  12  16:39  ..
-rw-r--r--@ 1 admin  staff  6.0K  10  12  17:13  .DS_Store
drwxr-xr-x@ 18 admin  staff  576B  9   17  04:39  放心课.app
→ Payload
```

通过上图可以发现，里面没有包含这样的一个 SwiftSupport 文件夹，说明虽然我们上传的 ipa包里面包含 SwiftSupport 文件夹，但是苹果应该给移除掉了，目的应该是为了减少ipa包下载大小。

但是打的包，为啥要包含这个 SwiftSupport 呢，带着这个疑问，就去查下了资料

<https://stackoverflow.com/questions/27894574/ipa-generated-by-swift-is-so-big-about-5mb>

```
The libraries containing the entire Swift language have to be embedded in the IPA. Those libraries are part of the app, not part of the system - because Swift has to work even with backwards compatibility, in part because it is constantly changing (independently of system updates), and in part in order to work on iOS 7 (where the system has never heard of Swift). And they are about 5MB in size.
```

由于Swift语言的不稳定性，为了向后兼容，IPA包需要包含完整的Swift动态库。但是从App Store下载的包为啥没有包含完整的Swift动态库，说明和系统有关，因为我的手机系统是 iOS15，下载的ipa包被苹果处理了，经测试发现当最低版本为iOS 12.2时候，打的包是不包含 SwiftSupport 这样的一个文件夹的

修改最低支持的系统为 iOS12 后，ipa包大小为**71.8MB**，减少了大概**52MB**左右

名称	修改日期	大小	种类
放心课.ipa	今天 15:44	71.8 MB	iOS 软件包归档
DistributionSummary.plist	今天 15:44	52 KB	Property List
ExportOptions.plist	今天 15:44	700 字节	Property List
Packaging.log	今天 15:44	1.7 MB	日志文件

解压ipa后，发现里面多了个 Symbol 文件夹

名称	修改日期	大小	种类
> Payload	今天 15:43	--	文件夹
> Symbols	今天 15:44	--	文件夹

这是由于在打包的时候，勾选了 Upload your app's symbols 选项，由于我们接入的有自己的崩溃日志平台，所以不需要苹果给我们提供崩溃日志，所以可以不选中该选项

App Store Connect distribution options:

Upload your app's symbols

Apple may use these symbols for purposes of providing you with symbolicated crash logs and other diagnostic information, compatibility testing of your app with Apple products and services, and for finding and fixing bugs and issues in Apple products and services and/or your app.

Manage Version and Build Number

This will change the version and build number of all content in your app to 1.4.0 (1).



Cancel

Previous

Next

名称	修改日期	大小	种类
放心课.ipa	今天 15:54	43.1 MB	iOS 软件包归档
DistributionSummary.plist	今天 15:55	52 KB	Property List
ExportOptions.plist	今天 15:55	701 字节	Property List
Packaging.log	今天 15:54	717 KB	日志文件

经过这样的改进，最终把放心课ipa包大小，由144.9MB改成了43.1MB

当把该ipa包上传到Testflight后，包大小却不是43.1MB，反而变大了

设备类型	下载大小 ?	安装大小 ?
通用	65.3 MB	105 MB

优化之前的ipa包，在Testflight上的大小

设备类型	下载大小 ?	安装大小 ?
通用	81.5 MB	149 MB

虽然安装大小减少了**40MB**，下载大小减少了**15MB**，但是和想象中差别很大

**说明：**安装大小其他相当于解压后的ipa包大小，也就是在App Store上展示的包大小，下载大小是用户下载的包大小，不同设备上会有差异，比如优化后的包在 iPhone 12 设备上大小如下

iPhone 12	30.1 MB	88.5 MB
iPhone 12 Pro	30.1 MB	88.5 MB

解压优化后的ipa包后大小，发现光Framework文件夹就占了89M，里面都是必须要依赖的一些第三方库，且都已经对这些库裁剪为只有 arm64 指令集，所以对Framework文件夹下的三方库已没啥可优化的空间了

```
→ 放心课 du -shc * | sort -rh
97M    total
89M    Frameworks
4.4M   放心课
2.4M   Assets.car
496K   LaunchScreen1@3x.png
244K   LaunchScreen1@2x.png
136K   _CodeSignature
92K    vconsole.js
68K    GTCaptcha4.bundle
24K    Base.lproj
12K    embedded.mobileprovision
8.0K   AppIcon76x76@2x~ipad.png
8.0K   AppIcon60x60@2x.png
4.0K   dealMCallCompatible.js
4.0K   PkgInfo
4.0K   Info.plist
4.0K   DripPay.bundle
```

## 可执行文件部分的优化

**符号表的裁剪：**对于 App 的主二进制而言，对外是不需要暴露符号信息的。而对外暴露的符号名称对 App 整体安全来讲也存在一些风险。通常通过设置 `STRIP_STYLE= all` 来裁剪所有符号。但通过：

```
objdump -exports-trie /path/to/MyApp.app/MyApp
```

还能获取到可执行文件中的符号，这部分可以通过设置 `EXPORTED_SYMBOLS_FILE` 为一个空文件解决 `EXPORTED_SYMBOLS_FILE=/path/to/emptyfile.txt`

对于自己构建的以及依赖的第三方动态库。只需要保留未定义的符号以及全局的符号其他的都可以去除。通常设置 `STRIP_STYLE=non-global`。

裁剪ipa包中Framework文件夹中的动态库

```
# Type a script or drag a script file from your workspace to insert its
path.
echo "current app arch"
APP_PATH="${TARGET_BUILD_DIR}/${WRAPPER_NAME}"
APP_EXECUTABLE_NAME="$(echo $WRAPPER_NAME | cut -d '.' -f1)"
arch="$(lipo -info "$APP_PATH/$APP_EXECUTABLE_NAME" | rev | cut -d ':' -
f1 | rev)"
echo $arch
if ! [[ "$arch" == *"arm64"* ]]; then
    echo "current app is not arm64 arch"
    exit
fi

echo "Stripping frameworks"
cd "${BUILT_PRODUCTS_DIR}/${FRAMEWORKS_FOLDER_PATH}"
framework_path=$(pwd)

for file in $(find . -type f -perm +111); do
    # Skip non-dynamic libraries
    if ! [[ "$(file "$file")" == *"dynamically linked shared library"* ]];
then
        echo "Skip non-dynamic libraries:$file"
        continue
    fi
    extracted_file_path=$framework_path${file:1}
    echo "$extracted_file_path"
    strip -x $extracted_file_path
done
```

通过对符号表裁剪，在Testflight上显示的安装大小减少了**10MB**

设备类型	下载大小 ?	安装大小 ?
通用	60.8 MB	78.5 MB

**死代码裁剪：**在构建完成之后如果是 C、C++ 等静态的语言的代码、一些常量定义，如果发现没有被使用到将会被标记为 Dead code。开启 `DEAD_CODE_STRIP = YES` 这些 Dead code 将不会被打包到安装包中。在 LinkMap 这些符号也会被标记为 `<<dead>>`。

由于我们工程默认已开启了该配置，没做处理。

由于我们目前 `iOS12.2` 以下系统的用户占比很低，建议把我们最低支持的系统版本升级为 `iOS 12.2`，这样经过前面一系列优化我们ipa包安装大小减少了50%左右，用户可以更快下载到我们的App，给用户带来更好的体验

## 依赖关系分析

接入新版IM后，我们放心课App，对第三方库依赖关系大致如下：代码生成的图



移除IM后，放心课App包大小为**22MB**，对第三方库依赖关系大致如下：手动画的图

