Mach-O

- Mach-O是Mach object的缩写，是Mac/iOS上用于存储程序，库的标准格式
- 属于Mach-O格式的文件类型有

```
#define MH_OBJECT    0x1     /* relocatable object file */
#define MH_EXECUTE   0x2     /* demand paged executable file */
#define MH_FVMLIB    0x3     /* fixed VM shared library file */
#define MH_CORE      0x4     /* core file */
#define MH_PRELOAD   0x5     /* preloaded executable file */
#define MH_DYLIB     0x6     /* dynamically bound shared library */
#define MH_DYLINKER  0x7     /* dynamic link editor */
#define MH_BUNDLE    0x8     /* dynamically bound bundle file */
#define MH_DYLIB_STUB    0x9     /* shared library stub for static */
                     /*  linking only, no section contents */
#define MH_DSYM      0xa     /* companion file with only debug */
                     /*  sections */
#define MH_KEXT_BUNDLE   0xb     /* x86_64 kexts */
```

常见的Mach-O文件类型

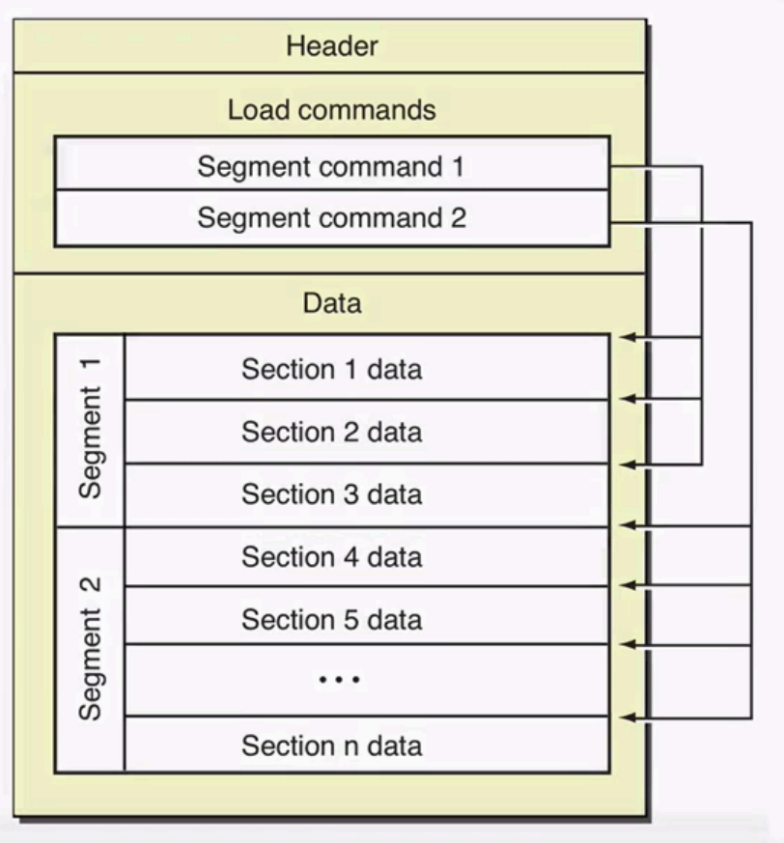
- MH_OBJECT
  - 目标文件（.o），c语言打包成可执行文件过程中的中间产物。即1.c 2.c -> 1.o 2.o -> 链接 -> 可执行文件
  - 静态库文件（.a），静态库其实就是N个.o文件合并在一起
- MH_EXECUTE
  - 可执行文件
- MH_DYLIB
  - 动态库文件
  - .dylib
  - .framework
- MH_DYLINKER
  - 动态链接编辑器
  - /usr/lib/dyld
- MH_DSYM
  - 存储着二进制文件符号信息的文件
  - 常用于分析app的崩溃信息

通用二进制文件 Universal Binary

- 通用二进制文件
- 同时适用于多种架构的二进制文件
- 包含了多种不同架构的独立的二进制文件

- 因为需要储存多种架构的代码，通用二进制文件通常比单一平台二进制的程序要大
- 由于两种架构有共同的一些资源，所以并不会达到单一版本的两倍之多
- 由于执行过程中，只调用一部分代码，运行起来也不需要额外的内存
- 因为文件比原来的要大，也被称为“胖二进制文件”（Fat Binary）

Mach-O的结构



dyld和Mach-O

- dyld用于加以下类型Mach-O
  - MH_EXECUTE
  - MH_DYLIB
  - MH_BUNDLE
- APP的可执行文件，动态库都是由dyld负责加载的

动态库共享缓存（dyld shared cache）

- 从iOS3.1开始，为了提高性能，绝大部分的系统动态库文件都打包存放到了一个缓存文件中（dyld shared cache）
- 缓存文件路径：/System/Library/Caches/com.apple.dyld/dyld_shared_cache_armX

- dyld_shared_cache_armX的X代表ARM处理器指令集架构
- v6
  - ✓ iPhone、iPhone3G
  - ✓ iPod Touch、iPod Touch2
- v7
  - ✓ iPhone3GS、iPhone4、iPhone4S
  - ✓ iPad、iPad2、iPad3(The New iPad)
  - ✓ iPad mini
  - ✓ iPod Touch3G、iPod Touch4、iPod Touch5
- v7s
  - ✓ iPhone5、iPhone5C
  - ✓ iPad4
- arm64
  - ✓ iPhone5S、iPhone6、iPhone6 Plus、iPhone6S、iPhone6S Plus
  - ✓ iPhoneSE、iPhone7、iPhone7 Plus、iPhone8、iPhone8 Plus、iPhoneX
  - ✓ iPad5、iPad Air、iPad Air2、iPad Pro、iPad Pro2
  - ✓ iPad mini with Retina display、iPad mini3、iPad mini4
  - ✓ iPod Touch6

- 所有指令集原则上都是向下兼容的
- 动态库共享缓存一个非常明显的好处是节省内存
- 现在的ida、Hopper反编译工具都可以识别动态库共享缓存