

iOS 性能优化篇 (启动时间)

参考文章: [优化 App 的启动时间](#)

[iOS启动时间优化](#)

[今日头条iOS客户端启动速度优化](#)

App运行原理

main() 执行前发生的事

- Mach-O 格式
- 虚拟内存基础
- Mach-O 二进制的加载

从 exec() 到 main()

改善启动时间

main阶段前

- 删除无用的代码（包括静态变量,类和方法）使用 Appcode对工程进行扫描，未使用的本地变量,未使用的参数,未使用的值
- 去除重名类方法,功能一致的方法等等.这些都会影响到App 启动时间.
- 如果可以的话尽量把 load 方法实现移动到 initialize。但是要注意 initialize 重复调用的问题.
- 减少不必要的 framework.

main阶段

总体原则无非就是减少启动的时候的步骤，以及每一步骤的时间消耗。

main阶段的优化大致有如下几个点：

- 主要针对 AppDelegate 里面的 ApplicationDidFinishLaunching 里面进行优化,一般我们会在这个代理方法里面做很多 类的初始化,还有一些相关业务的代码,我们要做的就是尽量让这些代码能更快速运行,或者如果能把其移动到项目启动之后在做更好,具体要参考具体情况. 比如友盟统计,埋点统计,项目缓存设置,初始化分享,初始化推送服务等等。 尽量能使用懒加载就使用懒加载,能放后台初始化就放到后台去初始化,不要卡主主线程,已经下线的业务要即时清理掉。
- 建议不要使用 +load 方法,由于 + load 方法是在 main() 函数之前调用,看到很多人习惯在这里方法用来实现单例实例化,虽然
- 底层页面控制器等使用 纯代码实现,而非xib或storyboard进行.因为 xib和storyboard 还需要重新解析成代码的方式来实现。
- 主UI框架tabBarController的viewDidLoad函数里, 去掉一些不必要的函数调用。
- 首次启动渲染的页面优化;

<http://lingyuncxb.com/2018/01/30/iOS%E5%90%AF%E5%8A%A8%E4%BC%98%E5%8C%96/>