

你一般是如何优化你的 APP 的?

一、首页启动速度

- 启动过程中做的事情越少越好（尽可能将多个接口合并）
- 不在 UI 线程上作耗时的操作（数据的处理在子线程进行，处理完通知主线程刷新）
- 在合适的时机开始后台任务（例如在用户指引节目就可以开始准备加载的数据）
- 尽量减小包的大小
- 优化方法：
 - 量化启动时间
 - 启动速度模块化
 - 辅助工具（友盟，听云，Flurry）

二、页面浏览速度

- json 的处理（iOS 自带的 `NSJSONSerialization`，`Jsonkit`，`SBJson`）
- 数据的分页（后端数据多的话，就要分页返回，例如网易新闻，或者 微博记录）
- 数据压缩（大数据也可以压缩返回，减少流量，加快反应速度）
- 内容缓存（例如网易新闻的最新新闻列表都是要缓存到本地，从本地加载，可以缓存到内存，或者数据库，根据情况而定）
- 延时加载 tab（比如 app 有 5 个 tab，可以先加载第一个要显示的 tab，其他的在显示时候加载，按需加载）
- 算法的优化（核心算法的优化，例如有些 app 有个 联系人姓名用汉语拼音的首字母排序）

三、操作流畅度优化：

- Tableview 优化（tableview cell 的加载优化）
- ViewController 加载优化（不同 view 之间的跳转，可以提前做好数据）

四、数据库的优化：

- 数据库设计上面的重构
- 查询语句的优化
- 分库分表（数据太多的时候，可以分不同的表或者库）

五、服务器端和客户端的交互优化：

- 客户端尽量减少请求
- 服务端尽量做多的逻辑处理
- 服务器端和客户端采取推拉结合的方式（可以利用一些同步机制）
- 通信协议的优化。（减少报文的大小）
- 电量使用优化（尽量不要使用后台运行）

六、非技术性能优化

- 产品设计的逻辑性（产品的设计一定要符合逻辑，或者逻辑尽量简单，否则会让程序员抓狂，有时候用了好大力气，才可以完成一个小小的逻辑设计问题）
- 界面交互的规范（每个模块的界面的交互尽量统一，符合操作习惯）
- 代码规范（这个可以隐形带来 app 性能的提高，比如用 `if else` 还是 `switch`，或者是用 `!` 还是 `==`）
- `code review`（坚持 `code Review` 持续重构代码。减少代码的逻辑复杂度）
- 日常交流（经常分享一些代码，或者逻辑处理中的坑）

[返回目录:全网各大厂 iOS 面试题-题集大全](#)

更多精选大厂 · iOS 面试题答案 PDF 文集



获取加小编的 iOS 技术交流圈：[937 194 184](#)，直接获取