



首页 > iOS开发

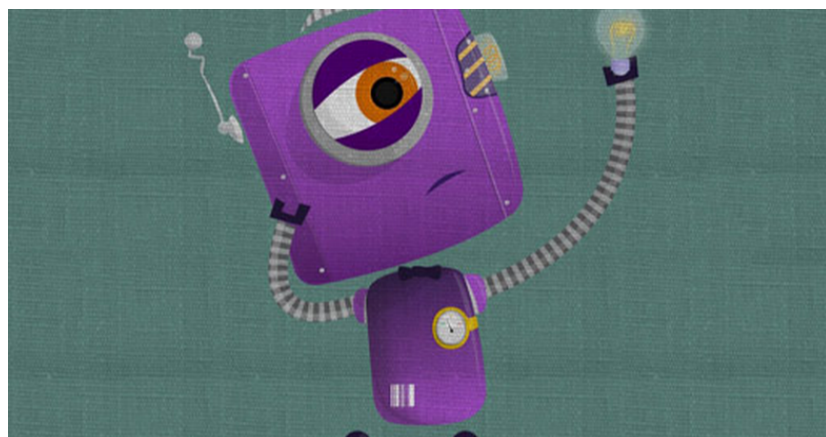
如何进行技术方案优化

2015-12-03 10:25 编辑: yishuiliunian 分类: iOS开发 来源: 一水的博客

2 951

程序员 需求 技术方案优化

招聘信息: Android开发工程师



本文是投稿文章, 作者: yishuiliunian

如何进行技术方案优化

首先这是一个很宏大的题目,我也只能窥其一斑。只说说自己的一些浅见和反思。因为之前有一段时间,在面对说要技术方案优化的时候,自己也有过一段不知所措的时期。后来才开始慢慢觉得自己找到点门道了,这篇文章也算是自己的一个总结吧。

什么是优化

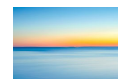
首先我们先破题,来谈谈“优化”这个事情。通常情况下,我们说到优化的时候,往往会伴随着对之前系统的吐槽。或是不好用,或是性能低,或是用起来很麻烦。巴拉巴拉。是的,当我们对原先的系统有槽点的时候,我们会谈到“优化”。而“优化”的前提也是,之前已经有过一个东西存在,而且真对目前的场景应景不再适合。这个是有需要对原有系统进行调整,以满足当前的场景与需求。那么所谓优化即是:对原有系统进行有目的的改造。

好吧,这听起来虽然说了什么,但其实什么都没说。因为这是一句大实话。

but,我们仔细分析一下,我们要进行优化必须能够:

1. 对原有系统的问题有所了解

热门资讯



如何让iOS保持界面流畅? 这些技巧你知道吗
点击量 10206



iOS数据库离线缓存思路和网路层封装
点击量 10084



如何设计一个iOS控件?(iOS控件完全解析)
点击量 9492



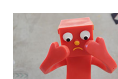
编程语言大牛王垠:编程的智慧,带你少走弯
点击量 8529



详解iOS上机题!附个人见解
点击量 8037



专业程序员必习:最牛B的编码套路
点击量 7296



培训机构毕业的程序员被歧视的背后逻辑
点击量 5365



地狱边境:从一张草图到收入破2000万美元
点击量 4794



快速搭建一个成熟,强壮的App框架
点击量 4725



【译】详细讲述iOS自定义转场
点击量 4385

综合评论

mark

中士圭 评论了 如何让iOS保持界面流畅? 这些技巧你知道吗...

没法用啊,效率低得要死要死
xuaswq1955 评论了 CIColor 酷炫动画指南...

Demo在哪里下载?
立白无糖口香糖 评论了 录音并发送的demo

虽然现在还没有机会考虑怎样留住客

- 2. 了解目前场景和需求
- 3. 有目的性的改造原有系统

我们来说一个我们通常会遇到的例子，也是在面试的时候会遇到的问题–“UITableView的性能优化”。其实每次有人问我这个问题，我内心都有千万只“草泥马”奔腾而过。没有具体的问题场景，只单单跑出来这样一个问题。是可以和他扯什么图片内存缓存了，避免圆角的使用了，预渲染，预加载了之类的东西。但是这些东西，真的对于在解决他们TableView卡顿的问题有效吗，不见得。套用《安娜卡列尼娜》一句话：

流畅的UITableView都是相似的，不流畅的UITableView各有各的不幸。

好了，吐槽到此为止。吐槽的目的是为了说明一点，你要进行优化，必须有一个特定的场景。在一个受限的范围内进行优化，因为这样目的是可控的。漫无边际的优化，和别人基于方法论的建议之类的东西，不一定对当前的问题有帮助。

比如，之前我们在做的一个社交类的App中，首页使用了UITableView，老板说怎么用着这么卡顿。然后我们就开始了“优化”。

首先，我们知道我们要优化的是第一个tab的tableview的滑动效率的问题。那总得有个监控的指标吧。对于程序猿来说，感觉这个不卡了，或者感觉这个卡，这个东西太模糊了。无法衡量啊。所以一定要量化。对于界面来讲就是大家常说的FPS，每秒帧率。于是我们测量了一下帧率，平均下来是25FPS。ou my god! 的确是有点卡。

然后我们知道对于ios来说如果能达到60FPS，那界面绝对不会有卡顿的感觉了。而很少有应用能达到这个水准。那么我们给自己设置了一个目标45FPS。btw,这个目标只是个阶段性目标。

好了下面的过程，就是朝着这个目标前进了。当然我们知道，造成FPS较低的原因，一般都是主线程做了太多的事情，导致帧率降低。这只是个大方向。而对我们来讲，我们需要精准的知道，主线程都做了些什么事情，导致帧率降低。

首先，我们发现的是，读取图片IO的过程发生在了主线程。IO过程一般是比较耗时的，于是我们像把该过程移到了后台线程中处理。发现帧率能够提高到33FPS，这还不够啊。革命尚未完成，同志仍需努力。

之后的过程中，我们把布局预处理，还有圆角，数据预加载之类的事情做上去之后，终于基本达到45。阶段性目标完成。

好了这是一个优化的例子：始于发现问题，止于目标达成。而重要的是其过程，描述问题！！！！

分析问题 (定性or定量)

其实，我一直比较坚信一句话：当你能够准确的描述一个问题的时候，你到解决问题就没剩几步了。比如刚才说的卡顿的问题，我们当时是这么描述的：图片读取发生了主线程，主线程中有一部分CPU片段用于文件读取和图片解码，造成主线程阻塞，从而导致帧率下降。当描述到这里的时候，解决方案就比较显而易见了，挪呗。搞到其他线程中之行。把主线程空出来。

而上面的这个描述还只是一个定性的描述分析。只是阐述了现象。虽然能够解决了一个问题，但是对整体问题的贡献有多大，也未可知。所以我们可以当时完全可以这样描述：我们图片缓存在文件系统的平均大小是1MB，其读取时间为10.7ms，图片格式为jpeg，解码一个1M的图片耗时是60ms，而我们知道60FPS，每帧给主线程用来处理任务的CPU时间为17.7S，也就说这个地方占用了大量CPU时间片来处理图片读与解码操作，从而造成了CPU阻塞，造成帧率没有达到60ms。

当我们使用定量的描述的时候，我们能够比较精确的知道，一个小问题，对于大问题来说到底意味着什么。而定量分

有灵性的小狐狸 评论了 为什么你的App 没人用？请按这8条逐一对照...

总算看明白了，做动画又多了一条路
xuaswq1955 评论了 CIColor 酷炫动画指南...

mark
codefucker 评论了 如何设计一个 iOS 控件?(iOS 控件完全解析...

感谢分享
zjshangren 评论了 Swift学习: 从 Objective-C到Swi...

我支持你啊，多出点demo
315909433 评论了 SpriteKit在iOS 9和 OS X 10.1...

= - = 花个万把 块钱 能不满意么
dushandz 评论了 权威组织： MacBook 可靠性、满意度继续遥遥领先...

mark
WzhGoSky 评论了 对 Xcode 菜单选项的详细探索（干货） ...

相关帖子

AFNetworking 如何请求有空格的网址

刷脸APP

Surferman - 极地冲浪

请问如何像今日头条的新闻详情一样，点击图片 调取photobrowser!

zsh配置文件消失

在xcode里已上传了App Store，然后在 itunes connect里要多久才能显示？

iOSjson解析：在son中嵌套有HTML怎么解析

我被猪滚了！相亲认识的男友占了我便宜，现在要分手

Xcode打开崩溃

析的方案中，当然包含了很多更多的细节信息，尤其是数据信息。这些也正是定量分析的优势所在。BUT，定量分析是一个非常耗时耗力的事情，你要拿到这么多的数据，你势必付出很多时间，在采集这些数据上面。对于app开发来讲，除非公司给了足够的资源（尤其是时间），你才能像个研究者一样去采集这些数据，一般情况是，大概都会止步到定性分析这一步。其实这也是看具体问题而定了。

不过无论你是使用定性分析的方式还是定量分析的方式。我们的目标是为了找到能够准确表述问题的方式，并且定位问题，以求找到解决方案。而为了达到这个目的一般情况下我们可以使用两种方式：

1. 你的编程功底和对iOS的了解程度都很深，那么完全可以从一些原理性的事情上去分析。我们称之为：逻辑分析法。
2. 或许你的编程功底很深，或许很浅，或许你尝试分析而没有结果。那么可以使用改改代码试试的方法了。我们称之为：实验法。

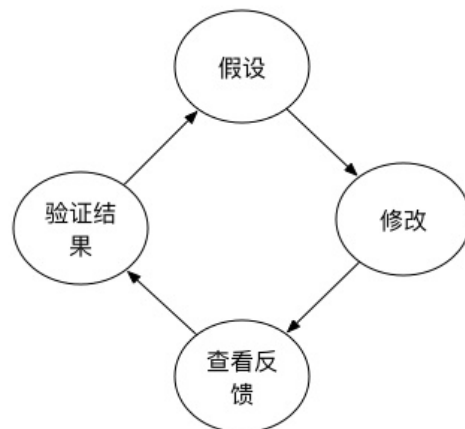
逻辑分析法，原理性分析

哈哈，套用马哲的一句话：事物是普遍联系的。既然是普遍联系的，不说必然存在因果，那么通过一定的逻辑分析。是可以找到他们之间的一些蛛丝马迹的关联的。这些关联或许可以解释一些什么。比如刚才卡顿的问题：原理就是主线程CPU被消耗过多，无法及时处理UI任务导致的。这只是一个例子。

我们进行逻辑分析的目的，是为了找到我们的某些代码和问题之间的因果性联系。就是说，我们能够明确知道造成UI卡顿的问题，就是因为IO的问题之类。这个话题说起来，比较深邃了。其中绝大部分实践的方法可以从《数理逻辑》这本书中找到。不过这是本讲数学的书，咱们得稍微换下脑子，把其中的定理，在编程中应用一下。因为我也只是意会了其中的某些东西，讲出来还没有那么功底。就只能麻烦各位自己去琢磨了。：)

实验法,Assume-Action-Response-Test-Assume

我称这个过称为AARTA。这是一个一直往复的过程，在分析的过程中，你得一次次的重复这个过程来找到真正问题的所在。其实，这个方法比较常应用在改BUG这个场景上。其实如果从广义上讲，按照上面咱们对技术优化的定义，改bug也算是一种优化。只不过这个场景比价特殊而已。当无法准确的分析原理，或者当前程序的复杂性过高（低内聚高耦合）已经超出人脑的计算能力范围的时候，那么就可以“猜”了。



(1) 假设 assume

根据以往的经验来，设定一个和问题域相关的假设。比如UI卡顿的问题，你怀疑是不是因为图片的问题呢。那么现在就假设是图片的问题！

(2) 尝试进行修改 action

微博



CocoaChina

加关注

【对 Xcode 菜单选项的详细探索（干货）】本文调研Xcode的版本是 7.1，基本是探索了菜单的每一个按钮。虽然从xcode4一直用到了xcode7，但是一般都只是用了一些基础的功能，说来也惭愧。http://t.cn/RUFVylt



12月4日 20:30 转发(52) | 评论(25)

【如何利用数据来支撑设计？】我最早的时候认为设计就是如何去做出各种新奇的图形、质感和界面，追逐潮流和创意。可是后来发现设计最难的是平衡各方面的因素，在条条框框的



既然假设是图片的问题，那么就把UIImageView从Cell上删掉吧。

(3) 看程序的反馈 Response

重新运行一遍程序，看一下程序运行的效果。FPS是否有所改善，而且改善的幅度有多大。

(4) Test

根据，程序的反馈和我们预先设定的目标来判断一下，当前改动是否满足了我们设计的目标。如果有，那么你大概就找到了问题的一个原因。如果没有那么进行下一步。

(5) 重新提出假设 Assume

既然不是图片的问题，那么会不会是其他事情上耗费了CPU呢。比如布局样式的计算。那么重新假设是局部样式的问题。在执行 (2) 过程。

所谓实验，即是大胆假设，小心取证，如此往复，以求终解。

监控

上面只是进行了一些方法论的探讨。但是有一件事情，是若要优化一定要做的。那就是“监控”。

监控是个非常重要的东西。

监控是个非常重要的东西。

监控是个非常重要的东西。

重要的事情说三遍。尤其是对于运行在生产环境的程序。这就像是一个体检，你得实时掌控程序的运行情况，知道问题出在了哪里，甚至有些时候知道：哎呀，出问题了。没有监控，程序一旦上线之后，就像脱缰的野马，跑到哪里，做了什么，你就是一头忙乱了。突然有一天，老板说有人反馈咱们的app经常崩溃，当你没有crash监控，这个你都不知道从哪里查起。

而且，监控也是优化的数据来源。他能够通过数据的指标来非常直观的告诉你，程序哪里有问题，你优化之后，效果是怎样的。现在网上有很多这方面的服务提供出来，比如bugly之类的，甚至有些是APM (application performance manager)，直接监控到程序的运行状态和性能。google一下，能搜出不少来。可以酌情，应用在自己开发的app中。

总结

说了半天，总结一下。优化是在可控的范围内有目的性的对现有程序的修改。一般可以使用逻辑分析法和实验法来定位、分析、描述问题。或者定性或者定量。无论哪种，要想优化，你得先建立起对自己app运行的监控体系。

欢迎关注iOS开发公共账号 iOS开发知识。





微信扫一扫

订阅每日移动开发及APP推广热点资讯

公众号: CocoaChina

我要投稿

收藏文章

分享到:

3

上一篇: 从copy和mutableCopy谈起

下一篇: 源码推荐(12.01B): 一行代码搞定自动布局, 自定义UICollectionViewFlowLayout

相关资讯

实现项目下载需求时遇过的那些坑

想要好代码问世 我们至少要多写两次

程序员都是工程师吗?

自己动手做用户测试 - 用最少的成本获取最大化的用户需

多些时间思考能少写些代码

让程序员变懒的工具: Jenkins + 蒲公英

你的公司会奖励代码量最大的那个人吗?

如何阅读大型代码库?

程序员的五种类型

如果40岁了还在写代码, 是一种幸福, 还是一种悲哀?



我来说两句



你怎么看? 快来评论一下吧!

发表评论

所有评论 (2)



beancafe

2015-12-04 01:48:36

并非干货

👍 0 💬 0 回复



黎明破晓残夜尽

2015-12-03 08:56:42

云里雾。

👍 0 💬 0 回复

[关于我们](#) [商务合作](#) [联系我们](#) [合作伙伴](#)

北京触控科技有限公司版权所有

©2015 Chukong Technologies, Inc.

京ICP备 11006519号 京ICP证 100954号 京公网安备11010502020289  京网文[2012]0426-138号