

21 | 除了 Cocoa, iOS还可以用哪些 GUI 框架开发？

18 | 怎么减少 App 电量消耗？

27 | 如何用 Flexbox 思路开发？跟自动布局比, Flexbox 好在哪？

34 | iOS 黑魔法 Runtime Method Swizzling 背后的原理

43 | 剖析使 App 具有动态化和热更新能力的方案

28 | 怎么应对各种富文本表现需求？

22 | 细说 iOS 响应式框架变迁, 哪些思想可以为我所用？

15 | 日志监控：怎样获取 App 中的全量日志？

29 | 如何在 iOS 中进行面向测试驱动开发和面向行为驱动开发？

25 | 怎样构建底层的发布和订阅事件总线？

02 | App 启动速度怎么做优化与监控？

16 | 性能监控：衡量 App 质量的那把尺

26 | 如何提高 JSON 解析的性能？

23 | 如何构造酷炫的物理效果和过场动画效果？

31 | iOS 开发学习资源和书单推荐

17 | 远超你想象的多线程的那些坑

01 | 建立你自己的 iOS 开发知识体系

04 | 项目大了人员多了, 架构怎么设计更合理？

03 | Auto Layout 是怎么进行自动布局的, 性能如何？

24 | A-B 测试：验证决策效果的利器

05 | 链接器：符号是怎么绑定到地址上的？

09 | 无侵入的埋点方案如何实现？

10 | 包大小：如何从资源和代码层面实现全方位瘦身？

12 | iOS 崩溃千奇百怪, 如何全面监控？

06 | App 如何通过注入动态库的方式实现极速编译调试？

07 | Clang、Infer 和 OCLint, 我们应该使用谁来做静态分析？

08 | 如何利用 Clang 为 App 提质？

Clang

场景

以前在工作中,有段时间连续发生了多次线上事故.在复盘时大家都提出是因为代码不规范、代码规范执行不到位.从而导致代码质量太差,无法监管.我们才被动处理线上事故。

会上牢骚发完,会后应该怎么执行呢? 我们都知道,监管手段是需要自己动手建设的.第三方工具无法满足所有的业务技术规范监控.在上篇文章“Clang、Infer 和 OCLint, 我们应该使用谁来做静态分析?”中,我们提到通过 Clang 提供的丰富接口功能就可以开发出静态分析工具,进而管控代码质量。

什么是Clang

08 | Clang 是什么? 它为什么叫 Clang? 它和 GCC 有什么关系?

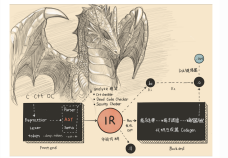


图 1 Clang 和 GCC 的关系图

Clang有优点

- 第一, 对于跨平台支持, Clang 相比 GCC 有优势, 因为它支持更多的平台, 比如 ARM、MIPS、PowerPC 等。
- 第二, 对于代码生成, Clang 比 GCC 更快速, 因为它使用了更先进的编译器技术, 比如 LLVM 的 JIT 编译器。
- 第三, Clang 支持更多的语言, 比如 C++11、C++14、C++17 等, 而 GCC 只支持 C++11。
- 第四, Clang 支持更多的平台, 比如 ARM、MIPS、PowerPC 等, 而 GCC 只支持 x86 和 ARM。
- 第五, Clang 支持更多的平台, 比如 ARM、MIPS、PowerPC 等, 而 GCC 只支持 x86 和 ARM。

Clang的作用

首先,Clang 会对代码进行词法分析,将代码切分成 Token

接下来,词法分析完后就会进行语法分析,将输出的 Token 先按照语法规则组合成语义,生成类似 VarDecl 这样的节点,然后将这些节点按照层级关系构成抽象语法树(AST)。

Clang的作用

语法检查工具 clang-check; 自动修复编译错误工具 clang-fixit; 自动代码格式化工具 clang-format; 新语言和新功能的迁移工具; 重构工具。