

MoonShim 项目申报书

项目名称	MoonShim	参赛方向	新项目申报（参与季度评选）
项目类别	开发工具 / 生态基础设施	GitHub ID	2227057182
当前版本	v0.1.0, 已发布至 Mooncakes	许可证	Apache-2.0

GitHub github.com/2227057182/moonshim Mooncakes mooncakes.io/docs/2227057182/moonshim

MoonShim 面向 MoonBit 库升级中的公共接口兼容问题。项目读取两个版本由 moon info 生成的 .mbti，识别 API 变化，并根据维护者确认的重命名规则生成可编译的兼容包装和类型别名。现有工具通常回答“发生了什么破坏性变化”，MoonShim 继续回答“旧调用方能否继续编译”。

真实需求与使用场景

- 库维护**：库作者发布升级版本时，把旧名称保留一个迁移周期，让已有调用方在兼容层上继续编译。
- 应用迁移**：应用维护者升级依赖时，先获得结构化差异和可编译适配器，再集中处理真正不兼容的部分。
- CI 门禁**：发布流程对比版本前后的 .mbti；没有明确规则的删除或签名变化阻止兼容性回退被静默合入。

核心功能与技术方案

解析器覆盖公开函数、方法、类型别名、结构体、枚举、Trait 和值，统一参数模式、默认值、返回类型与类型体。差异引擎区分新增、删除、重命名和签名变化；生成器只处理形状一致的重命名，为匿名参数赋予稳定名称，为方法恢复 self 接收者，并输出 Markdown 与 JSON 报告。无法证明安全的改动明确阻断。

原创性与生态定位

项目为原创 MoonBit 实现，不复制现有项目代码。与只做兼容性分析的库相比，MoonShim 的独立贡献是“规则约束下的代码生成 + 旧调用方编译证明”。它不自动推断两个函数语义等价，也不直接改写调用方源码；维护者必须确认重命名，项目只对机械化可支持的变化生成代码。

交付与验收证据

仓库包含 CLI、可复用库、迁移规则、示例报告、架构说明、开源许可证和端到端兼容夹具。当前 13 个自动化测试全部通过，并分别在 JS、Wasm、Wasm-GC 目标上验证；GitHub Actions 重复执行格式检查、严格检查、测试和生成稳定性检查。v0.1.0 已发布至 Mooncakes。

预期效果

- 缩短迁移周期**：维护者从两个接口快照得到结构化差异、人工报告和可编译兼容层。
- 提高发布可靠性**：没有明确规则的删除或签名变化会阻断发布流程，减少静默兼容性回退。
- 形成生态能力**：CLI 可用于发布流水线，核心库可被编辑器、包管理器和审计工具复用。

当前边界与后续计划

第一版聚焦显式函数和类型重命名。Trait 适配、运算符、自动迁移调用方和跨模块导入别名属于后续工作；无法证明语义的改动必须由维护者处理。该边界保证生成代码可解释、可测试、可维护，并为季度版本扩展留下清晰接口。