

PR #30585 完整报告

sgl-project/sglang

Enhance mechanical-refactor-verify skill with a whole-chain verifier, new relocation primitives, and generator inference

合并时间: 2026-07-14 16:45

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/30585>

执行摘要

- 一句话: 新增链验证器、重定位原语和生成器推断
- 推荐动作: 值得精读, 尤其是链验证器、分类规则和增量缓存的设计模式。对于构建自动化验证管线的团队有重要参考价值。

功能与动机

此前技能只能证明单次重定位 commit, 无法对整个重构分支 /PR 给出统一机器裁决。本 PR 补全了 split→construct→verify 的完整 workflow, 使 reviewer 可以信任整个 chain 而不必逐行审阅。

实现拆解

1. 全链验证器 (mechanical_refactor_reproduction_cli.py): 新增 CLI 入口, 遍历 base..branch 的每个 commit, 根据 commit message 中的分类词 (mechanical_provable / non_mechanical_provable) 分别调用证明脚本或标记为人工审查。支持并发运行 (--jobs) 和增量缓存 (--skip-passed)。
2. 扩展重定位原语 (reproduction_utils.py): 新增 move_assign (移动模块级常量)、extract_function (剪切内联块)、route_call_sites_through_field (路由调用至字段)、add/remove_imported_name (增减导入名) 等原语, 并增强 move_symbol 支持 after= 锚点和 leave_delegate 委托存根。
3. 生成器推断 (proof_generator.py): 新增 extract_function 推断 (识别从兄弟函数切出的内联块) 和类移动推断 (识别整个 ClassDef 的跨文件迁移), 并扩展对转发委托和常量伴随移动的推断。
4. 分类与验证契约: commit message 必须包含 exactly 一个分类词, 机器根据该词选择验证路径; non_mechanical_provable 声明必须诚实, 不得包含可证明的重定位 (向导中明确禁止)。
5. 测试覆盖: 添加了约 21 个测试文件, 覆盖链验证、分类、原语、缓存、报告生成、故障注入等场景。

关键文件:

- .claude/skills/mechanical-refactor-verify/scripts/mechanical_refactor_reproduction_cli.py (模块 验证工具; 类别 source; 类型 core-logic; 符号 ChainVerificationError,

CommitVerdict, ChainResult, verify_chain)：新增链验证器 CLI，是工作流的核心入口，负责分类、证明调度和报告生成。

- .claude/skills/mechanical-refactor-verify/scripts/mechanical_refactor_proof_generator.py (模块 生成器；类别 source；类型 core-logic；符号 _delegate_stub_attr, _next_sibling_assign_or_def, stmt_name, _stmt_symbol_name)：修改生成器，新增 extract_function 推断和类移动推断，扩展了可证明的重定位类型。
- .claude/skills/mechanical-refactor-verify/scripts/mechanical_refactor_reproduction_utils.py (模块 原语库；类别 source；类型 dependency-wiring；符号 _symbol_named, route_call_sites_through_field, op, predicate)：修改工具库，新增 move_assign、route_call_sites_through_field、add/remove_imported_name 等原语，并为 _find_unique_def 增加类匹配。
- .claude/skills/mechanical-refactor-verify/scripts/tests/proof_generator/test_infer_extract_functions.py (模块 测试集；类别 test；类型 test-coverage；符号 test_infer_extract_function_with_returned_local, test_infer_extract_function_keeps_leading_comment_in_body, test_infer_extract_function_no_return_text_when_body_is_whole_helper, test_infer_extract_function_edited_body_does_not_pass)：新增 extract_function 推断的测试用例，覆盖返回本地变量、保留前导注释、无 return text 等场景。
- .claude/skills/mechanical-refactor-verify/scripts/tests/reproduction_cli/test_verify_chain.py (模块 测试集；类别 test；类型 test-coverage；符号 test_chain_of_proved_and_declared_commits_passes, test_failing_proof_fails_the_commit_and_the_chain, test_proof_exiting_zero_without_a_pass_line_is_a_fail, test_main_exit_codes_reflect_the_chain_verdict)：新增链验证器的端到端测试，覆盖 PASS、FAIL、出口代码、设置错误等。

关键符号：verify_chain, render_report, infer_recipe, build_repro, move_symbol, move_assign, extract_function, route_call_sites_through_field, add_import, add_imported_name, remove_imported_name, _delegate_stub_attr, _next_sibling_assign_or_def, _symbol_named

关键源码片段

[.claude/skills/mechanical-refactor-verify/scripts/mechanical_refactor_reproduction_cli.py](#)

新增链验证器 CLI，是工作流的核心入口，负责分类、证明调度和报告生成。

```
# 关键类：CommitVerdict 和 ChainResult
# 每一个 commit 的验证结果，ok 表示 PASS 或 HUMAN_REVIEW
@dataclass(frozen=True)
class CommitVerdict:
    sha: str
    subject: str
    kind: str | None # 'mechanical_provable' 或 'non_mechanical_provable'
    verdict: str # PASS / FAIL / HUMAN_REVIEW 等
```

```
detail: str = ""
cached: bool = False

@property
def ok(self) -> bool:
    return self.verdict in (VERDICT_PASS, VERDICT_HUMAN_REVIEW)
```

```
@dataclass(frozen=True)
class ChainResult:
    base: str
    branch: str
    proof_dir: Path
    verdicts: list[CommitVerdict] = field(default_factory=list)

@property
def passed(self) -> bool:
    # 所有 commit 的 verdict 均为 ok 则认为 chain 通过
    return bool(self.verdicts) and all(v.ok for v in self.verdicts)
```

评论区精华

PR 作者在提交评论中指出，初始版本的单 commit 模式未校验 `extract_function`，导致生成正确脚本但报告 `UNSUPPORTED` 并错误退出。后续修复（`c73ca03`）增加了计数，并补充了委托、导入 / 赋值边界情况的测试覆盖。

- 单 commit 模式遗漏 `extract_function` 导致假失败 (`correctness`): 后续提交 `c73ca03` 修复了计数，并补充了委托和导入 / 赋值的边界情况测试。

风险与影响

- 风险：
 1. 证明脚本假阳性风险：如果 proof 脚本本身有 bug 但碰巧产生空 diff，验证可能错误通过。
 2. 缓存一致性问题：`--skip-passed` 基于内容哈希，若脚本正确性依赖环境（如 git 版本、formatter 版本），不同机器可能结果不同。
 3. 分类词误用：作者可能误用分类词（如将 `non_mechanical_provable` 用于纯重构），工具不做强制检查，依赖审阅者监督。- 影响：对用户无直接影响。对团队重构流程有显著改善：大规模重构（如 `model_runner` 拆分）可通过链验证器自动获得证明，减少人工审阅负担。测试框架和 CI 可集成验证步骤。- 风险标记：证明脚本假阳性风险，缓存环境依赖风险，分类词误用缺乏机器校验

关联脉络

- PR #31169 `Split initialize() into orchestration helpers`: 本 PR 提供的链验证器可用于验证该大规模重构的每个 commit，事实上 `test_infer_extract_functions` 中的示例用例即模拟了此类提取模式。

- PR #31168 Extract cuda-graph setup into a module: 同样属于 model_runner 拆分链, 本 PR 的 move_symbol 和 extract_function 原语可直接用于证明该重构。
- PR #31163 Extract per-architecture KV-cache pool builders into KVCacheConfigurator: 该 PR 的重构涉及大量跨文件移动和常量迁移, 本 PR 的 move_assign 和 move_symbol 原语可覆盖其验证需求。