

PR #1953 完整报告

THUDM/slime

[sglang_utils] flush_cache: log non-200 responses and back off before...

合并时间: 2026-05-27 13:52

原文链接: <http://prhub.com.cn/THUDM/slime/pull/1953>

执行摘要

- 一句话: 修复 flush_cache 非 200 响应时缺少日志和退避问题
- 推荐动作: 值得快速合并, 属于防御性改进。修复了静默错误和无效重试, 提升系统可观测性和稳定性。

功能与动机

/flush_cache 端点使用快照语义, scheduler 检查一次 is_fully_idle() 并返回 200 或 400。当有正在处理的请求时返回 400, 原有循环在非 200 时不休眠直接重试, 导致毫秒级高频重试, 并静默吞掉类似 “Another flush_cache is already in progress” 或 “Timed out waiting for idle state” 等错误。需要记录错误并增加休眠, 使重试退避有效、故障可观测。

实现拆解

1. 在 flush_cache 方法中, if response.status_code == 200 检查之后, 原先非 200 时直接继续循环, 现在添加一条日志语句 logger.info(f"Error flushing cache: HTTP {response.status_code} {response.text!r}") 记录非 200 响应状态码和正文。
2. 在日志后添加 time.sleep(1), 使每次非 200 响应后等待 1 秒再重试, 与 except 分支的行为保持一致。
3. 修改仅涉及 slime/backends/sglang_utils/sglang_engine.py 文件的 flush_cache 方法, 增加 2 行代码, 无其他文件变更。

关键文件:

- slime/backends/sglang_utils/sglang_engine.py (模块 后端引擎; 类别 source; 类型 core-logic): 核心变更文件, 修复 flush_cache 方法中非 200 响应的日志和退避逻辑。

关键符号: flush_cache

关键源码片段

slime/backends/sglang_utils/sglang_engine.py

核心变更文件, 修复 flush_cache 方法中非 200 响应的日志和退避逻辑。

```
def flush_cache(self):
    """Flush the cache of the server."""
    if self.node_rank != 0:
```

```
        return
    # flush cache will not return status_code 200 when there are pending requests
    for _ in range(60):
        try:
            response = requests.get(f"http://{self.server_host}:{self.server_port}/flush_cache")
            if response.status_code == 200:
                break
            # 非 200 响应：记录状态码和正文，然后退避 1 秒再重试
            logger.info(f"Error flushing cache: HTTP {response.status_code} {response.text!r}")
            time.sleep(1)
        except NewConnectionError as e:
            raise e
        except Exception as e:
            logger.info(f"Error flushing cache: {e}")
            time.sleep(1)
        continue
    else:
        raise TimeoutError("Timeout while flushing cache.")
```

评论区精华

无 review 评论，PR 由 zhuzilin 直接批准合并。讨论已在 PR body 中完整阐述：非 200 响应（如 400）未进入 `except` 分支，原循环无日志 / 退避；修复后统一行为。

- 暂无高价值评论线程

风险与影响

- 风险：变更极小，仅添加日志和休眠，风险极低。休眠 1 秒可能略微增加 `flush_cache` 总耗时（最多 60 秒），但已在 PR body 中说明这是有意为之的退避策略。无安全或兼容性问题。
- 影响：影响范围仅限于 `flush_cache` 调用方：非 200 响应时重试间隔从毫秒级变为 1 秒，整体等待时间更接近 60 秒预算；失败原因将出现在日志中，便于排查。不影响正常 200 情况。
- 风险标记：暂无

关联脉络

- 暂无明显关联 PR