# PR #21305 完整报告

sgl-project/sglang

Increase flush cache timeout in hicache CI

合并时间：2026-03-25 10:01

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/21305

## 执行摘要

本 PR 通过将缓存刷新重试机制从固定次数改为超时轮询，解决了 HiCache CI 中因异步操作导致的测试不稳定问题，提升了 CI 环境的可靠性。

## 功能与动机

PR 旨在修复 CI 测试中的缓存刷新失败问题。由于 HiCache 的异步操作（如 GPU↔Host↔L3 的写回）可能延迟调度器空闲检测，原重试逻辑在慢速 CI 环境下容易超时失败。变更动机参考了 GitHub Actions 运行失败记录和 PR #20746，直接针对此优化以提升 CI 稳定性。

## 实现拆解

改动集中于 python/sglang/test/test_utils.py 中的 flush_cache_with_retry 函数：

- 参数调整：将 retries 和 interval 替换为 timeout（默认 30.0 秒）和 poll_interval（默认 0.5 秒）。
- 逻辑重构：从固定次数循环改为基于当前时间与 deadline 的 while 轮询，持续尝试 POST 请求直到成功或超时。
- 文档更新：docstring 从“重试”改为“轮询”，并解释了短轮询间隔和长超时的设计意图。

```
def flush_cache_with_retry(base_url: str, timeout: float = 30.0, poll_interval: float = 0.5) -> bool:
    deadline = time.time() + timeout
    while time.time() < deadline:
        try:
            response = requests.post(f"{base_url}/flush_cache", timeout=10)
            if response.status_code == 200:
                return True
        except requests.RequestException:
            pass
        time.sleep(poll_interval)
    return False
```

## 评论区精华

在 review 中，gemini-code-assist[bot] 指出了关键缺陷：

> "The current timeout implementation can be violated, allowing the function to run for much longer than the specified timeout."

讨论强调请求调用时间未计入超时预算，可能导致显著超支。尽管问题被提出，但 PR 在批准后合并，未直接修复此缺陷，显示对 CI 稳定性的优先级权衡。

## 风险与影响

- 风险：超时控制不严格可能使测试运行时间超过预期，在测试环境中可能增加 CI 执行时间，但无生产系统影响。
- 影响：仅影响 CI 测试稳定性，通过更容忍慢速环境减少失败率，提升整体 CI 的可靠性，对用户和核心功能无直接影响。

## 关联脉络

本 PR 是近期 CI 优化系列的一部分：

- PR #20746（引用在动机中）可能涉及类似缓存或 CI 问题，提供演进背景。
- PR #21330 和 #21341 均聚焦 CI 测试流程改进，如启用 failfast 和添加健康检查，显示团队正系统提升测试基础设施的稳定性。