

PR #25990 完整报告

sgl-project/sclang

[CI] Enable nixl disaggregation test for decode radix cache

合并时间: 2026-05-21 19:29

原文链接: <http://prhub.com.cn/sgl-project/sclang/pull/25990>

执行摘要

- 一句话: 启用 nixl 解耦 decode radix cache 测试
- 推荐动作: 此 PR 值得关注, 因为它恢复了 nixl 后端的一项重要测试覆盖。对于使用 nixl 进行数据分片传输的团队, 建议关注此测试的稳定性。

功能与动机

之前在 PR#25145 中禁用了 nixl 后端的 decode radix cache 测试, 因为 nixl 后端不稳定。现在 bug 已在 PR#25699 中修复, 因此可以重新启用该测试来确保 nixl 后端下 decode radix cache 功能的正确性。

实现拆解

1. 修改测试文件 test/registered/distributed/test_disaggregation_decode_radix_cache.py 中 TestDisaggregationDecodeRadixCacheNixl 类的装饰器。
2. 将原本的 @unittest.skip("Temporarily disabled until nixl backend is stable.") 替换为 @unittest.skipUnless(is_in_ci() or _has_nixl(), "NIXL is required for decode radix cache disaggregation coverage.")。
3. 该变更使得测试在 CI 环境或存在 nixl 库时自动执行, 否则跳过。

关键文件:

- test/registered/distributed/test_disaggregation_decode_radix_cache.py (模块测试; 类别 test; 类型 test-coverage): 核心变更文件, 修改了测试类的启用条件, 使 nixl 后端测试重新生效。

关键符号: 未识别

关键源码片段

[test/registered/distributed/test_disaggregation_decode_radix_cache.py](#)

核心变更文件, 修改了测试类的启用条件, 使 nixl 后端测试重新生效。

```
# test/registered/distributed/test_disaggregation_decode_radix_cache.py
# 将原本无条件跳过的测试, 改为在 CI 或环境中存在 nixl 库时才执行
@unittest.skipUnless(
    is_in_ci() or _has_nixl(), # 当在 CI 中或系统安装了 nixl 时运行
```

```
"NIXL is required for decode radix cache disaggregation coverage.",
)
class TestDisaggregationDecodeRadixCacheNixl(
    DisaggregationDecodeRadixCacheTestMixin, PDDisaggregationServerBase
):
    transfer_backend_name = "nixl"
```

评论区精华

PR 没有 review 评论，但作者通过 `/rerun-test` 命令触发了 CI 测试，且测试通过。

- 暂无高价值评论线程

风险与影响

- 风险：风险较低。该 PR 仅修改测试的启用条件，不影响任何生产代码。潜在风险是如果 nixl 后端仍有未修复的 bug，测试可能失败并导致 CI 不稳定，但已有 bug 修复（PR#25699）。
- 影响：影响范围仅限于 CI 测试：nixl 后端的 decode radix cache 测试现在将在 CI 中运行，增加了测试覆盖。对用户和系统无直接影响。
- 风险标记：测试补强

关联脉络

- PR #25145 (unknown): 此 PR 中提及的禁用 nixl 测试的原始 PR。
- PR #25699 (unknown): 此 PR 中提及的修复 nixl 后端 bug 的 PR。