

PR #39585 完整报告

sgl-project/sglang

[NPU][CI] Scope NPU nightly artifact dirs by image; fix stale glm5_2 case

合并时间: 2026-09-15 22:55

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/39585>

执行摘要

- 一句话: NPU nightly 持久化目录按镜像隔离并修复 glm5_2 过期用例
- 推荐动作: 该 PR 值得快速浏览, 无需精读。值得关注的设计决策是: 在保持既有路径段数量契约的前提下, 用后缀方式把镜像身份注入持久化目录前缀, 避免了连锁改动; 同时用临时裁剪 job 的方式做定向验证后再回退, 是 CI 配置变更的一种稳妥实践。对后续维护者来说, 需要注意的是 sed 提取 tag 的假设条件, 以及在新增 NPU 镜像时保持该命名约定。

功能与动机

PR body 明确指出: NPU nightly 结果持久化到共享目录树, 但其前缀只由分支、工作流和运行元数据派生。当同一工作流内不同镜像 (CANN 9.0.0 vs CANN 9.1.0、a3 vs a5) 执行同一用例时, 两次运行解析到同一目录, 后一次运行会覆盖前一次的 metrics、log 和 plog artifacts, 导致结果无法按镜像归因。涉及多节点 POC 套件 (deepseek_v4_flash、glm_5_2、kimi_k3)。另外, CANN 9.1.0 多节点 mix 套件引用了已不存在的用例文件, job 在运行任何内容之前就会以 `The testcase does not exit` 中止。

实现拆解

1. 在 stage 模板中按镜像隔离持久化目录: `_npu-single-node-test-stage.yml` 与 `nightly-test-npu-e2e-multi-node.yml` 两处, 在完成 `workflow_name` 计算后追加 `--${image_label}`。image_label 通过 `sed 's#.*[:]##'` 从 `inputs.image` 提取 tag 部分 (丢弃 registry / repository 前缀), 例如 `cann9.0.0-a3`。采用后缀追加而非新增路径段, 是为了保持既有“按段数量解析”的消费者 (如 pod 侧 `RUN_LABEL` 派生逻辑) 继续工作。
2. pod 侧日志目录自动跟随隔离: 由于 pod 的 `RUN_LABEL` 由输出前缀派生, `logs/log` 和 `logs/plog` 也随之按镜像隔离, Python 与 shell 辅助函数零改动。legacy 布局与 `full-test-npu.yml` 保持不变。
3. 修复 CANN 9.1.0 mix 套件的过期用例引用: `nightly-test-npu.yml` 中 `glm_5_2` 的测试用例从已删除的 `test_npu_glm_5_2_w4a8_16p_gpqa.py` 切换为现存的 `test_npu_glm_5_2_w8a8_16p_gpqa.py`, 与 9.0.0 mix 套件保持一致, 消除 `The testcase does not exit` 的启动即失败。
4. 配套注释与自验证: 在 `python/sglang/test/ascend/e2e/test_npu_performance_utils.py` 中为 `DEEPSEEK_V4_FLASH_DEFAULT_MODEL_PATH` 补充“a5 用例默认模型”注释; 两个 stage 模板统一镜像 tag 追加逻辑的注释措辞。提交历史上可见作者曾临时裁剪 nightly job 以定向验证目录隔离效果, 验证后回退。

5. 测试与部署配套：不涉及模型或 kernel 代码，无推理路径变更，未新增单元测试；Accuracy 与 Speed 测试均不适用，CI 本身即验证手段。

关键文件：

- [.github/workflows/_npu-single-node-test-stage.yml](#)（模块 NPU CI；类别 infra；类型 infrastructure）：单节点 nightly 的 stage 模板，新增镜像 tag 到持久化目录前缀的核心改动点之一，直接影响 NPU 单节点 nightly 的产物归因。
- [.github/workflows/nightly-test-npu-e2e-multi-node.yml](#)（模块 NPU CI；类别 infra；类型 infrastructure）：多节点 POC 套件的模板，是双镜像（CANN 9.0.0/9.1.0）覆盖问题的主要受害场景，修复价值最高。
- [.github/workflows/nightly-test-npu.yml](#)（模块 NPU CI；类别 infra；类型 infrastructure）：修复 CANN 9.1.0 multi-node mix 套件引用已删除的 w4a8 用例，避免 job 启动即失败。
- [python/sglang/test/ascend/e2e/test_npu_performance_utils.py](#)（模块 性能测试；类别 test；类型 test-coverage）：为 DEEPSEEK_V4_FLASH_DEFAULT_MODEL_PATH 补充“a5 用例默认模型”注释，纯文档性改动，便于后续镜像 / 用例维护时理解常量用途。

关键符号：未识别

关键源码片段

[.github/workflows/_npu-single-node-test-stage.yml](#)

单节点 nightly 的 stage 模板，新增镜像 tag 到持久化目录前缀的核心改动点之一，直接影响 NPU 单节点 nightly 的产物归因。

```
# 从镜像输入中提取 tag，丢弃 registry / repository 前缀
# 例如 "registry.example.com/sglang/sglang:cann9.0.0-a3" -> "cann9.0.0-a3"
image_label=$(echo "${{ inputs.image }}" | sed 's#.*[:/]##')

# 将 image_label 后缀追加到 workflow_name 段（而非新增一个路径段），
# 使同一工作流内 CANN 9.0.0 与 CANN 9.1.0、a3 与 a5 的运行
# 持久化到互不覆盖的目录；同时保持既有“按段数量解析”的消费者兼容
workflow_name="${workflow_name}-${image_label}"
```

[.github/workflows/nightly-test-npu.yml](#)

修复 CANN 9.1.0 multi-node mix 套件引用已删除的 w4a8 用例，避免 job 启动即失败。

```
matrix:
  test_config:
    # 修复 CANN 9.1.0 multi-node mix 套件的过期引用：
    # 原 w4a8 用例文件已不存在，job 会以 "The testcase does not exit" 提前失败，
    # 现改用与 9.0.0 mix 套件一致的 w8a8 用例文件
    - name: glm_5_2_w8a8_16p_gpqa
      node_size: 2
      test_case: test/registered/npu/accuracy/glm5_2/test_npu_glm_5_2_w8a8_16p_gpqa.py
      test_type: 'accuracy'
```

评论区精华

本 PR 未产生实质性 review 讨论：sglang-npu-bot 直接批准且无评论，唯一的交互是机器人触发 `/tag-and-rerun-ci`。设计权衡主要由 PR body 与提交信息承载，关键决策是“镜像 tag 后缀追加到 workflow_name 段而非新增路径段”，以保证既有基于段数量的消费者（如 `RUN_LABEL` 派生与路径解析）无需改动；镜像 tag 提取使用 `sed 's#.*[:]##'` 丢弃 registry / repository 部分。这些约束说明作者对现有产物目录契约做过审慎核对。

- 暂无高价值评论线程

风险与影响

- 风险：
 - 目录命名变更的兼容性：任何按固定段数或固定格式解析持久化目录路径的下游脚本、看板或人工操作，需要适应 workflow_name--image_tag 的新命名；PR 用后缀而非新增段的方式刻意降低了这一风险，但仍存在未发现的硬编码解析器。
 - 镜像 tag 提取的健壮性：sed 's#.*[:]##' 依赖镜像名含 / 或 : 分隔符；若未来某镜像 tag 不含这些分隔符或格式变化，提取结果可能为空或异常，导致命名异常甚至目录回归到互相覆盖。
 - 覆盖范围有限：legacy 布局与 full-test-npu.yml 未改动，若这些工作流也存在同样的双镜像覆盖问题，本次未覆盖。
 - 用例路径有效性：glm_5_2_w8a8_16p_gpqa 用例目前存在于仓库，但如果该用例在 CANN 9.1.0 环境下不可用，job 仍可能失败；不过它已是 9.0.0 mix 套件在用的现成用例，风险较低。
 - 影响：影响范围集中在 NPU nightly CI 基础设施：持久化目录前缀新增镜像 tag 后，logs/log、logs/plog 与 metrics 产物可按镜像（CANN 9.0.0/9.1.0、a3/a5）隔离归因，多节点 POC 套件（deepseek_v4_flash、glm_5_2、kimi_k3）不再互相覆盖。对团队运维而言，查看 nightly 结果时需要习惯带镜像 tag 的目录命名；对用户与推理路径无任何影响。整体影响程度低到中，属于 CI 可观测性与可靠性的常规改进。
- 风险标记：artifacts 目录命名变更，镜像 tag 解析依赖 sed，仅覆盖 nightly 工作流

关联脉络

- PR #39411 [NPU] [CI] Fix multimodal_gen filter leakage and add daily scheduled run: 同属 NPU CI workflow (pr-test-npu.yml / nightly 配置) 治理线，体现 NPU CI 可靠性维护的延续，本 PR 是该方向上的又一次修复。
- PR #38632 [AMD][CI] Consolidate AMD workflows and retire ROCm 7.0 CI: 同类硬件专属 CI workflow 的归并与健壮性整理，与本 PR 一起反映了各硬件线持续治理 CI 配置的趋势，可相互参照。
- PR #39544 [misc] Trim redundant variants from the 8-gpu-h20 disaggregation test suite: 同为 CI 套件可靠性 / 效率调整（裁剪冗余变体），虽然属于 GPU 线，但与本 PR 一样以提升 CI 可维护性为目标。