PR #24369 完整报告

sgl-project/sglang

[Docker] fix: install nixl stub alongside nixl-cuXX binary

合并时间：2026-05-05 03:46

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/24369

执行摘要

- 一句话：Docker 中恢复 nixl stub 包安装
- 推荐动作：该 PR 修复了一个关键的回归问题，建议精读以了解 nixl 包的结构和安装最佳实践。

功能与动机

23593 因 nixl 轮子的无条件 nixl-cu12 依赖而移除了 nixl stub 包，但 sglang 代码仍从 nixl._api 导入，导致 HiCacheNixl 后端和 NIXL 解耦传输在 cu13 镜像上不可用。

实现拆解

1. 在 Dockerfile 的 CUDA 12 分支中，将安装命令从 pip install nixl-cu12 --no-deps 改为 pip install nixl nixl-cu12 --no-deps。
2. 在 CUDA 13 分支中，将安装命令从 pip install nixl-cu13 --no-deps 改为 pip install nixl nixl-cu13 --no-deps。
3. 使用 --no-deps 阻止 stub 包的无条件 nixl-cu12 依赖被安装，确保跨 CUDA 版本的二进制隔离。

关键文件：

- docker/Dockerfile（模块 部署脚本；类别 infra；类型 infrastructure）：唯一变更文件，在 cu12 和 cu13 的 pip install 命令中补充了 nixl stub 包。

关键符号：未识别

关键源码片段

docker/Dockerfile

唯一变更文件，在 cu12 和 cu13 的 pip install 命令中补充了 nixl stub 包。

```
# 安装 nixl 的 Python stub 包（提供 nixl._api 导入路径）
# 和对应 CUDA 版本的二进制包。使用 --no-deps 阻止 stub 的
# 无条件 Requires-Dist: nixl-cu12>=1.0.1 在 cu13 上被满足。
RUN --mount=type=cache,target=/root/.cache/pip \
```

```
if [ "${CUDA_VERSION%%.*}" = "12" ]; then \
    python3 -m pip install nixl nixl-cu12 --no-deps ; \
    python3 -m pip install cuda-python==12.9 ; \
elif [ "${CUDA_VERSION%%.*}" = "13" ]; then \
    python3 -m pip install nixl nixl-cu13 --no-deps ; \
    python3 -m pip install cuda-python==13.2.0 ; \
fi
```

评论区精华

gemini-code-assist[bot] 建议固定 nixl 和 nixl-cuXX 的版本为 1.0.1，并使用 && 替代 ; 以确保构建失败时立即终止。该建议未被采纳，PR 已合并。

- 建议固定版本并使用 && 替代 ; (infra): 建议未被采纳，作者未回复，PR 已合并。当前版本号和 ; 可能基于现有 Dcokerfile 风格保持一致。

风险与影响

- 风险：低风险，仅修改 Dockerfile 中的 pip 安装命令，且已在 cu12 和 cu13 镜像上端到端验证。潜在风险：未来 nixl 版本更新可能导致 stub 包行为变化，但当前变更只添加了显式包名，未引入不稳定因素。
- 影响：影响范围：Docker 镜像构建流程。影响程度：修复了 cu13 镜像上 HiCacheNixl 后端和 NIXL 解耦传输的缺失问题。用户无需额外操作，重建镜像即可生效。
- 风险标记：依赖包版本未固定

关联脉络

- PR #23593 [Docker] Prep for torch 2.11: cu129 fix, image validator, dep cleanup: 本 PR 是对 #23593 的修复，后者移除了 nixl stub 包导致导入失败。