

# PR #49763 完整报告

vllm-project/vllm

[ROCm][CI] Force native compile caches onto local disk

合并时间: 2026-07-25 12:14

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/49763>

## 执行摘要

- 一句话: 强制 AMD CI 编译缓存到本地磁盘
- 推荐动作: 该 PR 虽然代码变动极小 (10 行), 但背后修复了一个持续的 CI flaky 问题, 具有较高的工程价值。阅读者可关注 CI 脚本中环境变量的管理方式: 使用 `:=` 还是直接赋值, 取决于是否希望允许外部覆盖。对于 CI 内部路径, 直接赋值更可控。

## 功能与动机

AMD CI 构建作业 (如 Buildkite 上的 amd-ci) 频繁出现因编译缓存目录指向不可靠存储 (可能是 NFS 或内存不足的 tmpfs) 而导致的 flaky 失败。合并者 AndreasKaratzas 在 Issue 评论中明确指出该 PR "resolves flakes", 并附带了失败构建的链接。原参数扩展 `:=` 允许环境变量未设置时使用默认值, 但如果环境变量已被设置为不合适的路径, 则仍会使用错误路径。直接赋值消除了这种不确定性, 强制缓存写入本地磁盘的已知路径。

## 实现拆解

该 PR 仅修改了一个文件 `.buildkite/scripts/hardware_ci/run-amd-test.sh`, 具体改动分两部分:

1. 将缓存变量的参数扩展从 `:=` 改为直接赋值: 在 `initialize_native_environment()` 函数中, 原写法 `"${TORCHINDUCTOR_CACHE_DIR:=${native_root}/cache/torchinductor}"` 被替换为 `TORCHINDUCTOR_CACHE_DIR="${native_root}/cache/torchinductor"` 对 `TRITON_CACHE_DIR`、`VLLM_CACHE_ROOT`、`XDGE_CACHE_HOME` 三个变量同样处理。变化在于: 使用 `:=` 时, 如果环境变量已被外部设置, 则保留原值, 可能导致缓存路径指向非本地存储; 而直接赋值则始终覆盖为本地磁盘路径 `native_root` 下的子目录。
2. 增加调试日志: 在缓存目录创建并验证存在后, 增加一行 `echo` 输出 `VLLM_CACHE_ROOT` 和 `TORCHINDUCTOR_CACHE_DIR` 的实际值, 便于在 CI 日志中确认缓存位置正确。

关键文件:

- `.buildkite/scripts/hardware_ci/run-amd-test.sh` (模块 部署脚本; 类别 `infra`; 类型 `infrastructure`): 唯一的变更文件, 包含所有逻辑调整: 将编译缓存变量从带 fallback 的参数扩展改为直接赋值, 并增加日志输出。是整个 PR 的核心。

关键符号: `initialize_native_environment`

## 关键源码片段

### `.buildkite/scripts/hardware_ci/run-amd-test.sh`

唯一的变更文件，包含所有逻辑调整：将编译缓存变量从带 fallback 的参数扩展改为直接赋值，并增加日志输出。是整个 PR 的核心。

# `.buildkite/scripts/hardware_ci/run-amd-test.sh` ( 片段 )

```
initialize_native_environment() {
    local native_root="/tmp/vllm-native-${job_id}"
    # ... 其他路径定义

    # 强制编译缓存使用本地 /tmp 路径，而非从环境变量继承
    # 原写法为: "${VARIABLE:=default}"，允许外部覆盖，可能导致缓存指向 NFS 等不稳定存储
    TORCHINDUCTOR_CACHE_DIR="${native_root}/cache/torchinductor"
    TRITON_CACHE_DIR="${native_root}/cache/triton"
    VLLM_CACHE_ROOT="${native_root}/cache/vllm"
    XDG_CACHE_HOME="${native_root}/cache/xdg"

    # HuggingFace 缓存保持默认值，仍允许外部覆盖
    : "${HF_HOME:=/home/buildkite-agent/huggingface}"
    : "${HF_HUB_DOWNLOAD_TIMEOUT:=300}"
    : "${HF_HUB_ETAG_TIMEOUT:=60}"

    # 确保缓存目录存在
    mkdir -p "${TORCHINDUCTOR_CACHE_DIR}" \
        "${TRITON_CACHE_DIR}" \
        "${VLLM_CACHE_ROOT}" \
        "${XDG_CACHE_HOME}" \
        "${HF_HOME}" || return 1

    # 新增日志：输出关键缓存路径，便于在 CI 日志中定位问题
    echo "Native compile caches: VLLM_CACHE_ROOT=${VLLM_CACHE_ROOT}
    TORCHINDUCTOR_CACHE_DIR=${TORCHINDUCTOR_CACHE_DIR}"

    # ... 后续逻辑
}
```

## 评论区精华

该 PR 的审核评论非常少。claude[bot] 自动评论表示由于 PR 来自 fork，自动审核被禁用，需要维护者手动触发。合并者 AndreasKaratzas 在审核中仅给出 "LGTM" 并批准，未提出技术性讨论。因此没有暴露的设计争议或权衡讨论。

- 暂无高价值评论线程

## 风险与影响

- 风险：风险极低。变更仅涉及 CI 基础设施的 bash 脚本，不触及任何 Python 源码或运行时逻辑。主要风险是：如果 native\_root 路径不可写或磁盘空间不足，脚本后续的编译步骤可能会失败。但该路径在脚本同一函数中已被创建并检查（mkdir -p），且 CI 环境通常会清理 /tmp 目录，因此风险可控。没有安全、性能或兼容性风险。
- 影响：影响范围仅限于 AMD CI 构建环境，具体来说是调用 initialize\_native\_environment 的 CI 作业。受影响作业将获得更稳定的编译缓存行为，减少因缓存位置不当导致的 flaky 失败。对用户、产品、其他硬件平台无影响。影响程度中等（对 CI 稳定性有直接改善）。
- 风险标记：仅限于 CI 脚本，风险低

## 关联脉络

- PR #49739 [ROCm][CI] Wait for ROCm VRAM to settle between compiled and eager LL…: 同为修复 AMD CI 稳定性问题的 PR，且涉及编译测试与缓存相关逻辑。
- PR #49737 [ROCm][Docker] Drop MORI\_GPU\_ARCHS so MoRI autodetects the device arch: 同为 AMD CI 基础设施改进，移除硬编码 GPU 架构以修复 CI 崩溃。