

PR #43815 完整报告

vllm-project/vllm

[ROCm][CI] Stabilize Cargo cache and pre-test image checks

合并时间: 2026-05-28 11:24

原文链接: <http://prhub.com.cn/vllm-project/vllm/pull/43815>

执行摘要

- 一句话: 稳定 ROCm CI 的 Cargo 缓存和镜像检查
- 推荐动作: 值得参考其 CI 加固手法, 特别是缓存互斥锁和烟雾测试模式, 可推广至其他 CI 环境。

功能与动机

ROCm CI 中观察到并发 Cargo 缓存损坏, 且原有 Docker 清理逻辑可能因共享守护进程引发问题。需要不依赖全局缓存清理即可防止损坏, 并在单节点测试前提前发现镜像构建缺陷。

实现拆解

1. Dockerfile.rocm 缓存加锁与重试: 为 Cargo registry 和 git 缓存挂载添加 `sharing=locked` 和唯一 ID, 防止并行 BuildKit 作业写入冲突; 增加 `CARGO_NET_RETRY=10` 和 `RUSTUP_MAX_RETRIES=10` 提高网络容错。
2. amd.yaml 镜像烟雾测试: 在镜像构建步骤后插入 `docker run` 命令, 在容器内检查关键目录、可执行文件、工具链 (python3, uv, pytest) 以及 ROCm CLI, 并导入 torch/vllm 验证版本, 任何检查失败则提前退出。
3. run-amd-test.sh 清理策略与退出处理: 将 `cleanup_docker` 函数替换为只读的 `report_docker_usage`, 避免主动清理造成副作用; 重构 EXIT 陷阱, 确保容器清理不影响测试退出码。

关键文件:

- `.buildkite/hardware_tests/amd.yaml` (模块 测试编排; 类别 `test`; 类型 `test-coverage`): 核心变更: 在镜像构建后加入容器内烟雾测试, 提前发现镜像缺陷。
- `.buildkite/scripts/hardware_ci/run-amd-test.sh` (模块 CI 脚本; 类别 `infra`; 类型 `infrastructure`): 改进清理函数为安全报告模式, 并优化退出陷阱, 减少 CI 副作用。
- `docker/Dockerfile.rocm` (模块 镜像构建; 类别 `infra`; 类型 `infrastructure`): 核心缓存加锁与重试参数, 直接解决并发损坏问题。

关键符号: 未识别

评论区精华

PR 无 review 评论，仅获得维护者 tjtaanaa 的 LGTM。由于是 CI 基础设施变更，讨论可能在线下完成。

- 暂无高价值评论线程

风险与影响

- 风险：风险较低：缓存加锁可能在高并发时轻微阻塞，但远优于缓存损坏；烟雾测试增加约数秒构建时长；移除 Docker 清理可能导致磁盘使用积压，但报告模式已保留可见性，且避免误清理。
- 影响：影响限定在 ROCm CI 流程：镜像构建更可靠，单节点测试前有自动校验，Docker 资源清理改为被动观察。对 vLLM 核心逻辑无影响。
- 风险标记：Cargo 缓存竞争，镜像内容校验，Docker 清理策略变更

关联脉络

- 暂无明显关联 PR