PR #7606 完整报告

verl-project/verl

[env] fix: Update ascend image build workflow

合并时间：2026-08-31 09:03

原文链接：http://prhub.com.cn/verl-project/verl/pull/7606

执行摘要

- 一句话：Ascend 镜像改两阶段推送，规避 Quay GC 销毁中间镜像
- 推荐动作：建议镜像发布与 CI 维护同学精读该 PR：两阶段推送、digest 传递、防 GC tag、凭据隔离的组合是容器镜像 CI 的通用最佳实践。对一般训练框架使用者无需关注。后续可改进的点：将临时仓库迁移到组织级命名空间以消除单点依赖；为 merge 阶段补充 DIGESTS 为空时的显式失败断言；统一 a2/a3 的临时 tag 命名风格。

功能与动机

PR body 直接说明了根因：'The automatic garbage collection mechanism of the QUAY image registry causes the ARM image to be destroyed before the AMD image finishes building on Ascend.' 在原流程中，先完成构建的架构以 tmp-* 临时 tag 存放在正式仓库 quay.io/ascend/verl，等待另一架构构建期间会被 Quay GC 清理，导致多架构 manifest 合并时引用已失效的 digest。由于 ARM 与 AMD 构建耗时差异大，该问题表现为随机性的 CI 失败。

实现拆解

1. 新增临时仓库环境变量：两个 workflow 的 env 段各自新增 QUAY_TEMP_REPO: quay.io/yezib/verltest，与正式仓库 QUAY_REPO 分离，中间产物不再进入正式仓库。
2. 构建阶段改造为 digest 输出：build-push-digest 与 build-push-digest-v080 两个 job 中，docker/build-push-action@v6 由 tags: 直接写 tag 改为 outputs: type=image,name=...,push-by-digest=true,name-canonical=true,push=true，只向临时仓库推送 digest；随后新增 "Tag digest to prevent GC" 步骤，用 docker buildx imagetools create 立即为 digest 打 tmp-a2-*（a2）或 tmp-*（a3）临时 tag，并在注释中说明 "Quay.io 约 1 小时自动清理未打 tag 的 digest"。
3. 合并阶段区分源与目标仓库：merge job 新增 SOURCE_IMAGE=QUAY_TEMP_REPO、TARGET_IMAGE=QUAY_REPO 两个环境变量，digest 列表从临时仓库拼接，imagetools create 合并多架构 manifest 后写入正式仓库；注释提醒 " 临时仓库需为 Public，否则匿名读取 digest 会 401"，即 merge 阶段不依赖临时仓库凭据。
4. 清理阶段凭据隔离：删除临时 tag 的步骤单独用 QUAY_USERNAME_TMP / QUAY_PASSWORD_TMP 登录，通过 crane（go-containerregistry v0.20.2）执行 crane delete，与正式仓库凭据完全分离，删除失败以 || true 容忍。

5. 文档顺手清理：docs/ascend_tutorial/zh/dev_guide/performance/perf_tuning_on_ascend.rst 删除一行笔误的 ++actor_rollout_ref.ref.megatron.override_transformer_config.use_flash_attn=True 重复配置，与主变更无直接关系。
6. 测试与验证：无配套单测；验证依赖下次 Ascend 镜像构建实际运行，属于 "workflow 即测试 " 模式。

关键文件：

- .github/workflows/docker-build-ascend-a2.yml（模块 镜像构建；类别 infra；类型 infrastructure；符号 QUAY_TEMP_REPO, build_main）：Ascend a2 镜像构建主 workflow，承载两阶段推送核心改造（+65/-32），包括新增 QUAY_TEMP_REPO、push-by-digest 构建、防 GC tag、merge 与 cleanup 分离。
- .github/workflows/docker-build-ascend-a3.yml（模块 镜像构建；类别 infra；类型 infrastructure；符号 QUAY_TEMP_REPO, build_v080）：与 a2 同步的 Ascend a3 镜像构建 workflow（+65/-30），采用完全相同的两阶段推送方案，仅 Dockerfile 与临时 tag 命名不同。
- docs/ascend_tutorial/zh/dev_guide/performance/perf_tuning_on_ascend.rst（模块 文档；类别 docs；类型 documentation）：Ascend 性能调优文档，顺手删除一行笔误的 ref megatron use_flash_attn 重复配置（+0/-1），与主变更无直接关系。

关键符号：QUAY_TEMP_REPO, build_main, build_v080

评论区精华

该 PR 没有任何 review 评论，wucong25 直接 APPROVED，未产生讨论线程。值得注意的设计结论以代码注释形式固化在 workflow 中：临时仓库必须设为 Public，否则 merge 阶段匿名读取 digest 会返回 401；构建与合并阶段分别使用独立的仓库凭据，正式仓库与临时仓库的登录完全隔离。这两点是后续维护者最容易踩的坑。

- 暂无高价值评论线程

风险与影响

- 风险：
  1. 临时仓库依赖个人命名空间：quay.io/yezib/verltest 为个人账号仓库，构建链路强依赖其可用性、配额与凭据有效期；该 namespace 一旦被清理或凭据轮换且未同步更新 workflow，会导致发布中断。
  2. GC 策略差异未验证：方案假设临时仓库不会被 Quay GC 干扰，但 PR 未说明两个仓库的 GC 策略差异，若临时仓库同样触发 GC，竞态只是被转移而非消除。
  3. 临时 tag 清理容忍失败：crane delete || true 失败时不会告警，临时 tag 会逐渐累积，长期占用仓库配额。
  4. digest 传递依赖 runner 临时目录：merge job 通过 runner.temp 的 artifact 传递 digest 文件，上游失败时 DIGESTS 可能为空字符串，imagetools create 将失败，且没有重试机制。

5. 新 secrets 依赖：QUAY_USERNAME_TMP / QUAY_PASSWORD_TMP 若未在仓库 secrets 中配置，workflow 会在登录步骤直接失败。 - 影响：影响范围集中在 Ascend a2/a3 容器镜像的发布 CI 流程，不触碰任何训练、rollout 或模型代码路径。对镜像使用者完全无感：对外 tag 与镜像内容不变，只是中间产物的存放位置与推送时序改变。对团队而言，消除了发布流水线中的随机性失败，提升 Ascend 镜像发布的可靠性；同时引入了对个人 Quay 命名空间和新增 secrets 的运维依赖，需要团队内同步维护。该 " 两阶段推送 + 防 GC tag + 凭据隔离 " 的模式可复用到其他镜像仓库（如 Docker Hub 的自动清理场景）。 - 风险标记：依赖个人命名空间临时仓库，临时 tag 清理容忍失败，GC 策略差异未验证，夹带无关文档改动

关联脉络

- PR #7585 [ci] chore: Remove Ascend CI: 同为 Ascend CI 稳定性迭代：该 PR 临时注释不稳定的 Ascend 夜间 job，本 PR 修复镜像构建竞态，同属 Ascend 发布链维护。
- PR #7563 [doc] refactor: reorganize ascend_tutorial into zh/en directories: 该 PR 重构了 docs/ascend_tutorial 目录结构，本 PR 修改了该目录下的 perf_tuning_on_ascend.rst 文档。
- PR #7549 [ci] chore: correct step naming for Ascend ci: 同为 .github/workflows 下 Ascend CI workflow 的维护性修正，反映 Ascend CI 在近期处于高频调整期。