

PR #2854 完整报告

radixark/miles

[AMD] docker: retire the rocm700 variants, add a rocm10-mi35x variant

合并时间: 2026-09-09 13:37

原文链接: <http://prhub.com.cn/radixark/miles/pull/2854>

执行摘要

- 一句话: 淘汰旧 ROCm 7.0 镜像变体, 新增 ROCm 10 MI35x 构建以支持新硬件和 SDK。
- 推荐动作: 此 PR 是一个重要的基础设施更新, 旨在保持构建环境的现代化和可维护性。虽然没有涉及核心应用逻辑, 但对于支持 AMD 硬件平台至关重要。建议相关硬件或 CI 维护人员精读, 特别是 docker/Dockerfile.rocm 中的变更, 以理解新的构建流程和依赖管理策略。对于其他开发者, 了解新的构建变体即可。

功能与动机

上游 SGLang ROCm 7.0 基础镜像已停止构建, 而 ROCm 10 MI35X 基础镜像已每日构建并基于 Python 3.12 和 wheel SDK。同时, ROCm 10 修复了 ROCm 7.2 中存在的 VMM-pause 和 IPC-export 内存问题, 因此旧的 ROCr 和 legacy-IPC 变通方案不再需要。PR body 明确指出此变更是为了‘retire the rocm700 variants’并‘add a rocm10-mi35x variant’。

实现拆解

1. Docker 构建配置更新: 在 docker/build.py 和 docker/Dockerfile.rocm 中, 移除了 rocm700-mi35x 和 rocm700-mi30x 变体配置, 并新增 rocm10-mi35x 变体。新变体使用 rocm/sgl-dev:v0.5.18-rocm10-mi35x-20260831 基础镜像, 并设置了新的构建参数, 如 WHEELS_TAG_ROCM=rocm10-gfx950-v0.5.18、APEX_USE_PREBUILT=1 和 NVRX_INSTALL=1。
2. ROCm 10 特定依赖与修复: 在 docker/Dockerfile.rocm 中, 为 rocm10-mi35x 变体添加了多个关键步骤:
 - 安装来自 wheels shelf 的预编译 ROCm Apex 轮子, 以支持 Megatron 的融合操作 (APEX_USE_PREBUILT 控制)。
 - 安装 nvidia-resiliency-ext, 该扩展需要 glibc>=2.39, 仅在 ROCm 10 变体上启用 (NVRX_INSTALL 控制)。
 - 将 torch_memory_saver 固定到上游已合并的修复 VMM 分配粒度问题的提交 (06caa534), 并启用 --no-build-isolation。
 - 动态解析 tile_kernels 补丁路径, 以支持 Python 3.10/3.12。
 - 添加一个构建步骤, 通过创建符号链接来合并 ROCm wheel SDK 中重复的共享库, 以避免加载重复库导致的崩溃 (响应 ROCm/TheRock#7818)。

3. CI 与文档同步：更新 .github/workflows/docker-build.yml 的构建矩阵，移除旧的 ROCm 7.0 变体并添加新的 rocm10-mi35x。同步更新 docs/developer/ci/02-docker-build.md 和 docs/developer/versions.md 中的文档，描述新的变体列表和构建流程。
4. 移除过时配置：从 docker/Dockerfile.rocm 中移除了不再需要的 SGLANG_USE_ROCM700A 构建参数和环境变量，并移除了强制使用 numpy<2 的全局 pin。

关键文件：

- docker/Dockerfile.rocm（模块 构建定义；类别 infra；类型 infrastructure）：核心构建定义文件，包含移除旧变体、新增 ROCm 10 变体、集成 Apex/NVRx/torch_memory_saver 以及解决共享库重复问题的所有关键逻辑。
- docker/build.py（模块 构建脚本；类别 infra；类型 configuration）：构建变体配置的 Python 入口，定义了所有可用的构建目标和参数。移除了旧变体并添加了新变体定义。
- .github/workflows/docker-build.yml（模块 CI 工作流；类别 infra；类型 configuration）：CI 工作流定义，将构建变体的变更同步到 GitHub Actions。
- docs/developer/ci/02-docker-build.md（模块 构建文档；类别 docs；类型 documentation）：主要构建文档，详细描述了构建流程、变体和标签策略，需要与代码变更同步。
- docs/developer/versions.md（模块 版本文档；类别 docs；类型 documentation）：版本说明文档，列出了所有可用的 Docker 镜像变体，需要同步更新。

关键符号：未识别

关键源码片段

docker/Dockerfile.rocm

核心构建定义文件，包含移除旧变体、新增 ROCm 10 变体、集成 Apex/NVRx/torch_memory_saver 以及解决共享库重复问题的所有关键逻辑。

```
# 新增 ROCm Apex 预编译轮子安装，由 APEX_USE_PREBUILT 参数控制。
# Megatron 的 fused_weight_gradient_mlp_cuda 依赖此轮子。
RUN if [ "${APEX_USE_PREBUILT}" = "1" ]; then \
    pip install cxxfilt==0.3.0 py-cpuinfo==9.0.0 && \
    pip install --no-deps /tmp/wheels/apex-*.whl && \
    python3 -c "import fused_weight_gradient_mlp_cuda"; \
else \
    echo "[apex] prebuilt wheel skipped (APEX_USE_PREBUILT=${APEX_USE_PREBUILT})"; \
fi

# 动态解析 tile_kernels 补丁路径，以兼容 Python 3.10 和 3.12。
RUN pip install --no-deps tile_kernels==1.0.0 && \
    patch -p1 -d "$(python3 -c 'import site; print(site.getsitepackages()[0])')" < /tmp/amd_patch/tile_kernels.patch && \
    patch -p1 -d /opt/tilelang/src/tl_templates/hip < /tmp/amd_patch/tilelang_hip_fp8.patch

# 安装已合并上游修复的 torch_memory_saver，解决 ROCm 10 上的 VMM 分配崩溃。
# 使用 --no-build-isolation 以兼容 wheel SDK。
```

```
RUN pip install git+https://github.com/fzyzcjy/torch_memory_saver.  
git@06caa534822c5b980b61733a5d79ac731f9a5f9c --no-cache-dir --force-reinstall --no-build-  
isolation
```

合并 ROCm wheel SDK 中重复的共享库，避免加载重复库导致 LLVM 选项重复注册崩溃。

响应 issue ROCm/TheRock#7818。

```
RUN set -eux; \  
  SP="$(python3 -c 'import site; print(site.getsitepackages()[0])')"; \  
  if [ -d "$SP/_rocm_sdk_devel/lib" ]; then \  
    for f in "$SP"/_rocm_sdk_devel/lib/*.so*; do \  
      { [ -f "$f" ] && [ ! -L "$f" ]; } || continue; \  
      for t in _rocm_sdk_core _rocm_sdk_libraries; do \  
        c="$SP/$t/lib/$(basename "$f")"; \  
        if [ -f "$c" ] && cmp -s "$f" "$c"; then \  
          ln -srf "$c" "$f"; \  
          echo "[dedup] $(basename "$f") -> $t"; \  
          break; \  
        fi; \  
      done; \  
    done; \  
  else \  
    echo "[dedup] no wheel-SDK trees, skipping"; \  
  fi
```

评论区精华

此 PR 没有实质性的公开讨论。审核者 [guapisolo](#) 直接批准了变更，表明变更内容（基础设施更新）已预先达成共识或清晰明了。唯一的评论来自自动化机器人 [claude\[bot\]](#)，提示因是 Fork PR 而禁用了自动审核。

- 暂无高价值评论线程

风险与影响

- 风险：
 1. 兼容性风险：移除 rocm700-* 变体可能影响仍依赖这些旧镜像的外部用户或 CI 流程。文档更新部分缓解了此风险。
 2. 新依赖风险：引入 nvidia-resiliency-ext 和预编译的 ROCm Apex 轮子增加了新的外部依赖，需要确保其版本兼容性和长期维护。
 3. 构建复杂性风险：为 rocm10-mi35x 变体添加了多个条件步骤（APEX_USE_PREBUILT、NVRX_INSTALL、库去重脚本），增加了 Dockerfile 的复杂性和潜在的维护负担。
 4. 上游依赖风险：PR 依赖于已合并的上游修复 (torch_memory_saver PR#109)。虽然当前已合入，但未来上游变更可能需要同步调整。
- 影响：影响主要限于 AMD ROCm 构建基础设施和文档。直接用户影响是：1) 使用 rocm700-mi35x 或 rocm700-mi30x 镜像的下游项目需要迁移到 rocm720-mi35x 或新的 rocm10-mi35x。2) 新的 rocm10-mi35x 镜像为 MI35x 硬件提供了基于最新 ROCm 10

SDK 的构建环境，可能提升性能和稳定性。对团队而言，此变更减少了维护过时构建变体的负担，并使 CI 矩阵与上游可用资源保持一致。

- 风险标记：旧硬件支持移除，新外部依赖引入，构建流程复杂化

关联脉络

- PR #3124 Bump sglang to v0.5.19: 同属基础设施更新，升级了 sglang 依赖版本，与本 PR 中 ROCm 10 基础镜像的 SGLang 版本选择有潜在关联（均为更新上游依赖）。