

PR #29621 完整报告

sgl-project/sglang

Extract reusable VMM shareable-handle helpers from register_graph_inputs

合并时间: 2026-07-02 05:02

原文链接: <http://prhub.com.cn/sgl-project/sglang/pull/29621>

执行摘要

- 一句话: 提取 VMM 可复用的跨进程句柄共享帮助函数
- 推荐动作: 建议精读 `export_shareable_handles` 和 `release_mappings` 的实现, 前者展示了 FABRIC->POSIX 优雅降级和全 rank 一致性保证, 后者演示了 partial undo 模式。此 PR 是 CUDA driver API 封装的良好范例。

功能与动机

为了减少重复代码, 使 VMM 句柄处理步骤可被其他需要跨 rank 显存共享的代码复用, 同时使 `register_graph_inputs` 更易读。如 PR body 所述: “Extracting them enables that reuse and makes `register_graph_inputs` easier to read.”

实现拆解

1. 文件重命名与函数提取: 将 `custom_all_reduce_vmm_utils.py` 重命名为 `vmm_utils.py`。原来内联在 `VmmGraphInputManager.register_graph_inputs` 中的句柄导出 (FABRIC/POSIX)、fd 交换、导入与映射逻辑全部提取为模块级函数。
2. 核心公用函数: 新增 `export_shareable_handles` (自动从 FABRIC 降级到 POSIX)、`exchange_posix_fds` (SCM_RIGHTS fd 交换)、`import_and_map_alloc`、`map_chunk_into_span`、`import_peer_handle`、`release_mappings` (安全释放)。将私有 `_check_drv` 改为公有 `check_drv`, 并添加 `make_rw_access_desc` (构造 `CUmemAccessDesc`) 和 `all_ranks_ok` (全 rank 一致性检查)。
3. `register_graph_inputs` 重构: `VmmGraphInputManager.register_graph_inputs` 依次调用上述函数, 先导出 handles, 再 all-gather 或交换 fds, 最后逐段映射。所有 `cuMem*` 调用序列和 flags 保持不变, 仅在共享时替换了一个错误字符串。
4. 测试套件: 新增 `test_vmm_utils.py`, 注册为 2-gpu-large 测试。通过 `_create_alloc` 创建 shareable VMM 分配, 测试 round-trip 导出 -> 交换 -> 导入映射, 覆盖 `transport=fabric` (仅当 NVLink 可用) 和 `posix`, 以及 `n_chunks=1, 3` 两种映射形状。

关键文件:

- `python/sglang/srt/distributed/device_communicators/vmm_utils.py` (模块 通信层; 类别 source; 类型 rename-or-move; 符号 `check_drv`, `make_rw_access_desc`, `all_ranks_ok`, `release_mappings`) : 核心文件: 包含所有提取的可复用 VMM 句柄函数, 如 `export_shareable_handles`、`exchange_posix_fds` 等; 文件从

custom_all_reduce_vmm_utils.py 重命名而来。

- test/registered/unit/distributed/test_vmm_utils.py (模块 分布式测试; 类别 test; 类型 test-coverage; 符号 _gloo_group, _make_prop, _fabric_available, _create_alloc) : 新增测试文件: 覆盖 export/import round-trip, 包括两种 transport 和两种 chunk 数量, 确保重构后逻辑正确。
- python/sglang/srt/distributed/device_communicators/custom_all_reduce_v2.py (模块 通信层; 类别 source; 类型 dependency-wiring) : 仅修改导入路径, 指向新的 vmm_utils 模块。

关键符号: check_drv, make_rw_access_desc, all_ranks_ok, release_mappings, export_shareable_handles, exchange_posix_fds, import_and_map_alloc, map_chunk_into_span, import_peer_handle, recv_loop

评论区精华

无公开 review 讨论。维护者 @merrymercy 直接批准。PR 描述中明确声明“no behavior change”。

- 暂无高价值评论线程

风险与影响

- 风险: 主要风险为文件重命名和函数公有化可能影响未发现的引用。但仓库内仅 custom_all_reduce_v2.py 一处导入, 且 CI 测试覆盖了所有受影响路径 (含 expandable_segments)。此外, CUDA 驱动调用序列未经修改, 测试验证了跨进程 round-trip, 故风险较低。潜在风险包括其他分支或未合并 PR 对新模块路径的依赖, 但可通过 CI 自动检测。
- 影响: 对最终用户无影响 (无行为变化)。对开发者: vmm_utils.py 现为共享句柄处理的标准工具库, 未来 MoE、HiCache、DS V4 等模块可复用。重构降低了 register_graph_inputs 的复杂度 (减少 inline 代码约 120 行), 并提供了完善的单元测试, 降低了后续改动引入缺陷的风险。
- 风险标记: 文件重命名可能导致旧导入中断, 函数公有化可能影响内部封装, CUDA 驱动调用序列未变, 风险较低

关联脉络

- 暂无明显关联 PR