PR #32829 完整报告

sgl-project/sglang

[CI] Graceful teardown for kv_canary and EAGLE spec fixtures

合并时间：2026-08-02 16:10

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/32829

执行摘要

- 一句话：测试夹具改优雅关闭，消除 GPU 残留误报
- 推荐动作：值得精读。这是一个罕见的 " 机制导向 "CI 修复案例：作者没有止步于表面修好，而是用定量 A/B 证伪自己的初次结论，最终给出 " 减少 38% 但不消除 " 的诚实评估。对维护大型 GPU 测试集群的团队，terminate_and_kill_process_tree 加静态反汇编扫描的方法论可借鉴；同时它揭示了 SIGKILL 关停 CUDA 服务器时 pinned host 内存回收的耗时机制，对理解测试隔离设计有参考价值。

功能与动机

两次 CI 失败（test_self_e2e_perturb_real_kv_unused_cache.py 在 GPU-idle 30s 预检查超时、test_spec_eagle.py 的 TestEagle3Overlap 报 kill_process_tree 60s 未回收）均源于 fixture 用裸 SIGKILL 关停服务器：teardown 在 0.01s 内返回但 52 GiB（单服务器）或 295 GiB（四卡）内存仍被分配，是否在下个类的 30s gate 内回收完全取决于内核回收速度。与 #31746 / #31871 同根因：大 pinned host KV 池的 cudaHostUnregister 与内核回收极慢，实测 243 GB pinned host 池在裸 SIGKILL 后需 25.94 s 才完全干净。PR body 还指出全仓库 502 个杀掉服务器进程树的 teardown 中，只有来自 #31746 / #31871 的两个做了优雅关闭，因此需要系统性推广。

实现拆解

1. 提取公共 teardown 工具：在 python/sglang/test/test_utils.py 新增 terminate_and_kill_process_tree(process, terminate_timeout=60, **kill_kwargs)，序列为 SIGTERM → 最多等 60s → kill_process_tree（SIGKILL）兜底，并把 wait_timeout 等关键字透传给后者的 kill 等待逻辑。
2. 收敛本 PR 触发失败的夹具：python/sglang/test/kv_canary/e2e_base.py 的 CapturedServerE2EBase（CanaryE2EBase 基类）与 python/sglang/test/server_fixtures/spec_eagle_fixture.py 的 SpecEagleServerBase，这两个是 run 30448736325 中分别在 GPU-idle 30s 超时和 SIGKILL 后 60s 未回收的现场。
3. 推广到其余单服务器夹具：default_fixture（约 26 个测试模块）、chunked_prefill_test_utils（约 12 个）、streaming_session_fixture、pcg_spec_fixture、ngram_fixture、mmmu_fixture、dsa_mtp_fixture、standalone_fixture、hybrid_attn_backend_fixture、eagle_fixture、vlm_utils、test_deterministic_utils。其中

default_fixture 与 eagle_fixture 原本就传 wait_timeout=60，语义从 "SIGKILL 后等60s" 变成 "SIGTERM 后再 SIGKILL 并等 60s"；kl_mamba 测试的 HiCache L2/L3 原本有内联的 terminate + wait 序列，现在删除重复代码统一走公共函数。

4. 修复连锁 import 回归：vlm_utils 不再 import kill_process_tree 后，test/registered/vlm/test_vision_openai_server_a.py（通过 from sglang.test.vlm_utils import * 间接依赖）与 test/registered/xpu/test_gemma_4_e2b.py（直接 from sglang.test.vlm_utils import kill_process_tree）会分别触发 NameError / ImportError；PR 将这两个文件的 teardown 也一并转换为 terminate_and_kill_process_tree。
5. 有意保留的例外：disaggregation_fixture 因涉及三个进程加活跃的 mooncake 传输引擎，SIGTERM 行为需单独 PR 验证；ascend/xpu 专用 fixtures 因非 CUDA runner、GPU-idle gate 不适用，未纳入本次修改。
6. 验证手段：本地 1×H100 与 4×H100 devbox 跑全部转换后的 fixture（含 TestEagle3Overlap、4×H100 TP4 的 kl_mamba 12 个用例）全部通过；另用静态方法反汇编 522 个受影响测试模块中的 3284 个 setUp/tearDown 方法排查不可解析的 globals，提前捕获了上述两个 import 回归。

关键文件：

- python/sglang/test/test_utils.py（模块 测试工具；类别 test；类型 test-utility；符号 terminate_and_kill_process_tree）：新增 terminate_and_kill_process_tree()，是全部转换夹具共用的 teardown 工具：先 SIGTERM 等待最多 60s，再 SIGKILL 兜底，让服务器在用户态释放 CUDA 上下文与 pinned host 内存。
- test/registered/radix_cache/unified_radix_tree/test_unified_radix_cache_kl_mamba.py（模块 缓存测试；类别 test；类型 test-coverage）：#31871 的延续：三个类（普通、HiCache L2、HiCache L3）teardown 统一收敛到公共函数，删除各自内联的 terminate + wait 重复代码；其中 HiCache L2/L3 正是 pinned host 池回收最慢的场景。
- python/sglang/test/server_fixtures/default_fixture.py（模块 默认夹具；类别 test；类型 test-coverage；符号 DefaultServerBase）：影响面最大的公共夹具（约 26 个测试模块继承），teardown 从 kill_process_tree(pid, wait_timeout=60) 升级为带 SIGTERM 前置的版本。
- python/sglang/test/kv_canary/e2e_base.py（模块 端到端基座；类别 test；类型 test-coverage；符号 CapturedServerE2EBase, CanaryE2EBase）：本 PR 的两个 CI 失败现场之一（test_self_e2e_perturb_real_kv_unused_cache.py 的 GPU-idle 30s 超时），CanaryE2EBase 所在基类 CapturedServerE2EBase 的 teardown 被转换。
- python/sglang/test/server_fixtures/spec_eagle_fixture.py（模块 推测解码；类别 test；类型 test-coverage；符号 SpecEagleServerBase）：另一个失败现场（TestEagle3Overlap 的 kill_process_tree 60s 未回收），SpecEagleServerBase teardown 被转换。

关键符号：terminate_and_kill_process_tree

关键源码片段

python/sglang/test/test_utils.py

新增 terminate_and_kill_process_tree()，是全部转换夹具共用的 teardown 工具：先 SIGTERM 等待最多 60s，再 SIGKILL 兜底，让服务器在用户态释放 CUDA 上下文与 pinned host 内存。

```
def terminate_and_kill_process_tree(
    process,
    terminate_timeout: float = 60,
    **kill_kwargs,
) -> None:
    """先优雅关闭服务器，再对剩余进程树 SIGKILL 兜底。

    直接 kill_process_tree 时，内核需要在进程回收阶段解构 CUDA 上下文、
    解钉 host 内存；实测大 pinned 池（243 GB 量级）会让 GPU 残留约 26 秒，
    足以让下一个测试类的 GPU-idle 预检查（30s gate）误报失败。
    先发 SIGTERM，让服务器在用户态 unregister 这些资源，再走 SIGKILL。
    """
    # 第一步：SIGTERM 请求优雅退出，最多等 terminate_timeout 秒
    process.terminate()
    try:
        process.wait(timeout=terminate_timeout)
    except subprocess.TimeoutExpired:
        pass # 超时不阻塞，直接进入下面的 SIGKILL 兜底
    # 第二步：无论是否优雅退出，都确保整棵进程树被清掉
    kill_process_tree(process.pid, **kill_kwargs)
```

test/registered/radix_cache/unified_radix_tree/test_unified_radix_cache_kl_mamba.py

31871 的延续：三个类（普通、HiCache L2、HiCache L3）teardown 统一收敛到公共函数，删除各自内联的 terminate + wait 重复代码；其中 HiCache L2/L3 正是 pinned host 池回收最慢的场景。

```
    @classmethod
    def tearDownClass(cls):
        # HiCache L2 场景：服务器持有 243 GB 量级的 pinned host KV 池，
        # 裸 SIGKILL 时内核回收曾让下一个测试类的 GPU-idle 预检查超时；
        # 先 SIGTERM 让用户态 unregister，实测将残留窗口从约 25.94 s
        # 缩短到 16.01 s（减少约 38%），仍未完全消除但显著降低误报概率
        terminate_and_kill_process_tree(cls.process, wait_timeout=60)
```

评论区精华

最有价值的讨论是 alisonshao 在 issue 评论中对结论的两次修正：

- 第一次 A/B 只测了 device memory（裸 kill 0.01s 返回、52/295 GiB 未释放；优雅关闭 5.4-6.0s、约 0 残留），作者随后承认该测量没有覆盖真正的失败机制。

- 换上真实 pinned host pool 复测（Llama-3.1-8B、--enable-hierarchical-cache --hicache-ratio 4.0，243 GB pinned host + 67.5 GB device）：裸 kill 后完全回收需 25.94 s，优雅关闭后仍需 16.01 s。作者总结："This is the mechanism. The fix reduces the next class's exposure by ~38% but does not eliminate it."
- 另有一个 follow-up 建议：抽象 test base 如 _PerturbRealKvUnusedCacheBase 仍会被 unittest 从 __main__收集，在最后一次真实 teardown 后空跑一次 GPU-idle gate 才 SkipTest，移出测试模块可去掉这个多余检查点。
- 优雅关闭 vs 裸 SIGKILL 的 GPU 内存回收 A/B 验证 (testing): 转换后的 fixture 全部验证通过；但作者随后承认首次测量只覆盖了 device memory，未覆盖真实失败机制。
- 根因校正：pinned host KV 池的回收耗时才是关键 (correctness): 接受作者结论：机制确认，修复是缓解而非根治；总回收时间约 26 s 在繁忙共享 runner 上仍可能超过 30 s gate。
- 抽象 test base 仍被 unittest 收集并空跑 GPU-idle gate (design): 留作 follow-up，不在本 PR 处理。
- 分组测试下的 rerun-test 行为待验证 (question): 未在本 PR 处理，作者标记为待办。

风险与影响

- 风险：
  1. GPU 残留窗口未完全消除：作者实测优雅关闭后仍需约 16 s 才完全干净，在繁忙共享 runner 上仍可能逼近 30 s gate，只是概率降低。
  2. SIGTERM 不响应的服务器会让每个 teardown 先等满 terminate_timeout 60s，再叠加 kill 阶段的 wait_timeout，最坏情况每个类多出最多约 120 s，CI 总时长可能上升。
  3. vlm_utils 不再 re-export kill_process_tree 属于符号级破坏性变更，任何在其外部依赖该符号的测试或工具代码会立即 ImportError（本 PR 修了仓库内两个，仓库外测试可能受影响）。
  4. PD disaggregation 路径未覆盖：ChunkedTestPDBase 复用 PDDisaggregationServerBase.tearDownClass()，仍为裸 SIGKILL，相关场景的 GPU 残留问题依旧存在。
  5. 涉及约 100 个测试模块的 teardown 行为变更，虽经静态扫描验证，但个别 fixture 可能有自己的进程管理假设，需要 CI 多轮观察。 - 影响：对 CI 稳定性：直接消除两类失败现场（GPU-idle 30s 超时、SIGKILL 后 60s 未回收），并把约 100 个测试模块的 teardown 从 " 依赖内核回收 " 改为 " 用户态释放优先 "，降低跨测试类污染（失败的类不再毒化下一个类的 gate）。对成本：每次 teardown 增加数秒，CI 全量耗时上升，但换来的是更稳定的调度。对团队：确立了 " 先 SIGTERM 再 SIGKILL" 的 teardown 标准模式，后续新 fixture 应直接使用公共工具而非各自内联序列。 - 风险标记：GPU 残留窗口未完全消除，SIGTERM 超时可能拖慢 CI，vlm_utils 符号移除引发连锁，disaggregation 路径未覆盖，涉及约 100 个测试模块

关联脉络

- PR #31746 [Fix] Release hierarchical cache host pool on graceful shutdown: 同根因链条的起点：修复 hicache 主机 KV 池在优雅关闭路径的 unregister；本 PR 的 SIGTERM

前置依赖该用户态释放路径存在。

- PR #31871 [CI] Graceful teardown in kl_mamba hicache tests to release pinned host pool: 首次内联 terminate-then-kill 序列的 PR；本 PR 将其提取为公共函数 terminate_and_kill_process_tree 并推广到全部单服务器夹具。
- PR #31509 per-class GPU-idle gate（PR body 引用）: GPU-idle 预检查机制是本 PR 要消除的误报来源；PR body 明确引用该 gate（#31509）作为失败触发条件。