PR #35407 完整报告

sgl-project/sglang

[CI] Trim the base-c 4-gpu-h100 stage from 5 shards to 4

合并时间：2026-08-19 15:48

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/35407

执行摘要

- 一句话：base-c H100 CI 从 5 分片裁至 3，重排测试并清理死代码
- 推荐动作：值得 CI 维护者精读。本 PR 展示了基于 GPU 足迹与测试频率做 placement 的完整决策链：先算 max_parallel 量化收益，再逐文件核对真实 GPU 需求与 mock 依赖，最后用共享 mixin 消解跨文件复用冲突。需特别关注两点：TestMTP 删除带来的 dp2+MTP 覆盖缺口（建议后续在 extra 补回），以及新估 est_time 经 live partition model 校正后的分片稳定性。

功能与动机

PR body 明确指出：base-c-test-4-gpu-h100 runs its shards serially (max_parallel = 5 // 3 = 1)，因此削减 est_time 是线性墙钟收益；目标是从 5470s/16 文件降到 3572s/13 文件、5 shards -> 3，且 'Nothing is dropped from CI'。同时指出 'Most of it is placement: several tests sat on the 4-GPU runner without needing four GPUs'，并借机清理了两处无效的 skip 覆盖声明和一个自 2026-05-26 起长期 disabled 的测试文件。

实现拆解

1. 瓶颈定位与总体策略：base-c 阶段 5 个分片在 4-gpu-h100 runner 上串行执行，max_parallel 为 1，任何 est_time 削减都近似线性转化为墙钟收益。作者把策略定为 placement 优先：只把真正需要 4 卡且属于高频 per-commit 覆盖的用例留在 base-c，其余按 GPU 占用与测试频率下沉或外移。
2. mock-only 测试下沉：test/registered/unit/layers/test_flashinfer_comm_fusion.py 的 collective 全部通过 patch.object mock，三个 runner 跑的是相同 mock，因此从 4-gpu-h100 移到 base-b/1-gpu-small，并去掉 4-gpu-b200、4-gpu-gb300 两个注册；test/registered/hicache/test_hicache_storage_3fs_backend.py 最大 --tp-size 为 2 且两个 config 都是 H100，移到 base-b/2-gpu-large。
3. PD/PP 测试按 GPU 足迹拆分：test/registered/disaggregation/test_disaggregation_unified_memory.py 两个类为 prefill tp1 + decode_base_gpu_id=1，只需 2 卡，移到 2-gpu-large；test/registered/disaggregation/test_disaggregation_kimi_linear.py 剩余两类需要 3 卡留在 4-gpu。Gemma4 两个 tp1/pp2 类拆到新文件 test/registered/pp/test_pp_gemma4.py，注册到 base-b/2-gpu-large，test/registered/pp/test_pp_single_node.py 保留四类 tp2/pp2；两类都带 is_in_amd_ci() skip，AMD 注册不变。

4. 低频组合外移 extra-b：test/registered/ep/test_deepep_small.py 中 TestHybridDPTP、TestTBOWithTPAttn、TestTBOWithTPAttnAndDenseDP 三个类原样搬入新文件 test/registered/ep/test_deepep_small_extra.py（est_time=210，extra-b）；test/registered/models_e2e/test_qwen3_next_models.py 中两个 AllocFail 类搬入 test/registered/models_e2e/test_qwen3_next_models_extra.py（est_time=250，extra-b）。原生类体逐字节保留，唯一改动是 PD parity 基类改名以便 import。
5. 共享 mixin 提取：新增 python/sglang/test/kits/pd_parity_kit.py 提供 PDLogprobParityMixin（含 generate 与 test_logprob_parity），让 test_disaggregation_kimi_linear.py 与 test_disaggregation_unified_memory.py 共享实现，同时避免 unittest 把裸基类当测试收集。提交历史显示该 mixin 先放 server_fixtures 再移到 kits，最终保持 fixture 层只负责启动 server。
6. 死代码清理与注册修正：删除 test_deepep_small.py::TestNoGatherdBuffer（被 skip，经核实 test_deepep_large.py::TestDeepseek 覆盖同等组合）、TestMTP（原 skip 声明 "covered in TestMTPWithTBO" 不准确，实际是 tp4/dp2 vs tp4/dp4，删除会丢失 dp2+MTP 组合，作者选择接受），以及整个 test/registered/rl/test_release_memory_occupation.py（自 2026-05-26 起 disabled="Temporarily disabled - needs investigation"，无 import、同族自包含）。随后同步下调两个主文件的 est_time：test_deepep_small.py 478 -> 270，test_pp_single_node.py 500 -> 280。

测试 / 配置 / 部署配套：全部变更都是 CI 注册与测试文件位置调整，无源码、schema 或部署改动；base-b-test-2-gpu-large 吸收三个移动后 est_time 从 6701s 升至 8121s，分片数 5 -> 7，max_parallel 1 -> 2，串行波次 5 -> 4，需要观察 runner 稳定性。

关键文件：

- test/registered/pp/test_pp_gemma4.py（模块 PP 测试；类别 test；类型 test-coverage；符号 TestGemma4PPAccuracy, TestGemma4PLEPPAccuracy, test_gsm8k, test_mmmu）：把 Gemma4 两个 tp1/pp2 精度测试从 test_pp_single_node.py 拆出并注册到 2-gpu-large，是本次按 GPU 足迹重排的核心示例，且保留了完整的精度门禁注释。
- test/registered/ep/test_deepep_small_extra.py（模块 DeepEP 测试；类别 test；类型 test-coverage；符号 TestHybridDPTP, TestTBOWithTPAttn, TestTBOWithTPAttnAndDenseDP, test_gsm8k）：DeepEP 三个低频交叉验证类从 test_deepep_small.py 拆出并注册到 extra-b，避免占用 per-commit 的 base-c 时间，是本次覆盖频率分层的关键文件。
- test/registered/rl/test_release_memory_occupation.py（模块 RL 内存测试；类别 test；类型 deletion；符号 TestReleaseMemoryOccupation, _setup_engine, _common_test_params, _test_initial_generation）：长期 disabled（2026-05-26 起）的死代码，无任何 import，整体删除 477 行，释放 base-c 4-gpu-h100 的占位时间。
- python/sglang/test/kits/pd_parity_kit.py（模块 PD 工具；类别 test；类型 test-coverage；符号 PDLogprobParityMixin, generate, test_logprob_parity）：新增 PDLogprobParityMixin，把 PD logprob parity 测试基类抽成 mixin 供解聚测试文件复用，并避免 unittest 收集裸基类，是多个测试文件安全移动的前提。
- test/registered/ep/test_deepep_small.py（模块 DeepEP 测试；类别 test；类型 test-coverage；符号 TestHybridDPTP, TestNoGatherdBuffer, TestTBOWithTPAttn,

TestTBOWithTPAttnAndDenseDP）：删除 TestNoGatherdBuffer（skip 重复）、TestMTP（覆盖声明不准确）及两个 TBO 类，est_time 从 478s 降至 270s，是主文件裁剪的核心。

- test/registered/pp/test_pp_single_node.py（模块 PP 测试；类别 test；类型 test-coverage；符号 TestGemma4PPAccuracy, TestGemma4PLEPPAccuracy）：移除 Gemma4 两个类到新文件，est_time 从 500s 降至 280s，保留四类 tp2/pp2 用例，是 4-GPU 阶段瘦身的一部分。
- test/registered/disaggregation/test_disaggregation_unified_memory.py（模块 解聚测试；类别 test；类型 test-coverage；符号 TestUnifiedMemoryDisaggregation, test_logprob_parity）：两个 PD 类仅需 2 卡（prefill tp1 + decode_base_gpu_id=1），整体从 4-gpu-h100 移到 2-gpu-large，并复用 PDLogprobParityMixin。
- test/registered/models_e2e/test_qwen3_next_models_extra.py（模块 Qwen3 测试；类别 test；类型 test-coverage；符号 _make_args, TestQwen3NextLazyExtraBufferAllocFail, TestQwen3NextLazyExtraBufferLargePageAllocFail）：两个 AllocFail 类从 test_qwen3_next_models.py 拆到 extra-b，保留 KL 散度门禁配置的同时减轻 base-c 负担。

关键符号：PDLogprobParityMixin, generate, test_logprob_parity, TestGemma4PPAccuracy, TestGemma4PLEPPAccuracy, TestHybridDPTP, TestTBOWithTPAttn, TestTBOWithTPAttnAndDenseDP, TestQwen3NextLazyExtraBufferAllocFail, TestQwen3NextLazyExtraBufferLargePageAllocFail

关键源码片段

test/registered/ep/test_deepep_small_extra.py

DeepEP 三个低频交叉验证类从 test_deepep_small.py 拆出并注册到 extra-b，避免占用 per-commit 的 base-c 时间，是本次覆盖频率分层的关键文件。

```
import os
import unittest
from types import SimpleNamespace

from sglang.srt.utils import kill_process_tree
from sglang.test.ci.ci_register import register_cuda_ci
from sglang.test.run_eval import run_eval
from sglang.test.test_utils import (
    DEFAULT_MODEL_NAME_FOR_TEST_MLA,
    DEFAULT_TIMEOUT_FOR_SERVER_LAUNCH,
    DEFAULT_URL_FOR_TEST,
    CustomTestCase,
    popen_launch_server,
)

# 三组 DeepEP 交叉组合从 base-c 挪到 extra-b：低频验证不进 per-commit
# 关键路径，但每个维度仍保留至少一个代表用例。
```

```
register_cuda_ci(est_time=210, stage='extra-b', runner_config='4-gpu-h100')


class TestHybridDPTP(CustomTestCase):
    # hybrid DP/TP：tp=4、dp=2，验证 enable-dp-attention 下 MoE all-to-all
    # 走 deepep 后端，是 DP 注意力 + DeepEP 的组合覆盖。
    @classmethod
    def setUpClass(cls):
        cls.model = DEFAULT_MODEL_NAME_FOR_TEST_MLA
        cls.base_url = DEFAULT_URL_FOR_TEST
        cls.process = popen_launch_server(
            cls.model,
            cls.base_url,
            timeout=DEFAULT_TIMEOUT_FOR_SERVER_LAUNCH,
            other_args=[
                '--trust-remote-code',
                '--tp', '4',
                '--enable-dp-attention',
                '--dp', '2',
                '--moe-a2a-backend', 'deepep',
                '--cuda-graph-max-bs-decode', '128',
                '--max-running-requests', '256',
            ],
        )

    @classmethod
    def tearDownClass(cls):
        kill_process_tree(cls.process.pid)

    def test_gsm8k(self):
        args = SimpleNamespace(
            base_url=self.base_url,
            model=self.model,
            eval_name='gsm8k',
            api='completion',
            max_tokens=512,
            num_examples=200,
            num_threads=128,
        )
        metrics = run_eval(args)
        print(metrics)

        # 与源文件保持一致的精度门禁，确保迁移不改变用例语义。
        self.assertGreater(metrics['score'], 0.60)
```

评论区精华

该 PR 没有任何 reviewer 评论。Issue 评论中作者先后三次发起 /rerun-test 来验证移动后的 10 个测试文件，bot 回报在 4-gpu-h100、2-gpu-h100、1-gpu-5090 等不同 runner 上全部

通过，证明 placement 后的注册生效。最终由作者直接合并，无外部 review 交锋；但作者自己在 body 里留下了重要提示：DeepEP 拆分把 attention-TP + TBO 组合移出 per-commit 路径，且两个新 2-gpu-large 的 est_time（900s、220s）是估算值，需由 live partition model 后续校正。

- 移动后测试的 rerun 验证 (testing): bot 报告在 4-gpu-h100、2-gpu-h100、1-gpu-5090 等 runner 上全部通过，移动后的测试注册生效。

风险与影响

- 风险：
  1. per-commit 覆盖降级：DeepEP 的 attention-TP + TBO 组合从 base-c 移到 extra-b，每个 PR 的常规 CI 不再执行该组合，回归只能由 extra 或 nightly 捕获，这是 author 自述的已知取舍。
  2. 覆盖声明不准确导致净丢失：TestMTP 的删除基于 "covered in TestMTPWithTBO" 的声明，但作者核实该声明不准确（tp4/dp2 vs tp4/dp4），因此 dp2 + MTP 组合实际不再有任何覆盖。
  3. est_time 为估算值：新增 2-gpu-large 的 900s、220s 来自父文件拆分估算，live partition model 会在运行后校正，存在短时分片不均衡、max_parallel 计算变化的可能。
  4. 重排依赖 rerun 验证：移动文件通过多次 /rerun-test 验证，但 base-c 缩减到 3 分片后，若剩余用例出现长尾，可能重新失衡；test_pp_gemma4.py 的 test_mmmu 仍需 5-7 分钟手动运行，未计入自动门禁。
  5. 用户面影响为零：无产品代码、无 API 变更，风险全部局限在 CI 基础设施内部。 - 影响：对 CI 系统：base-c-test-4-gpu-h100 墙钟时间约减少 35%（5470s -> 3572s），开发者获得更快的提交反馈；base-b-test-2-gpu-large 负载上升，分片数从 5 增至 7，但 max_parallel 从 1 升到 2，串行波次反而从 5 降至 4。对团队：per-commit 对 attention-TP + TBO 等低频组合的即时覆盖被转移到 extra-b，团队需要依赖 extra/nightly 结果来兜底。对用户：无任何影响。 - 风险标记：per-commit 覆盖降级，删除覆盖声明不准确，est_time 估算待校正，测试重排依赖 rerun 验证

关联脉络

- PR #35467 [AMD][DI][CI] Run MI355X disagg nightly at 7AM UTC: 同属 CI 基础设施时间 / 资源调优方向（调整 nightly 调度时间），与本 PR 一起反映 CI 从串行满载走向分层调度的趋势。
- PR #35446 Fix HiCache PP sync test fixture: 同为 HiCache 测试文件修复 / 调整，本 PR 移动了 test_hicache_storage_3fs_backend.py 并处理过 HiCache PP KL 测试，属于同一测试域。
- PR #35298 [Fix] DCP: advertise the logical KV-event block size: disaggregation 相关修复，本 PR 重排了 test_disaggregation_unified_memory.py 与 test_disaggregation_kimi_linear.py，两者存在覆盖关联。