

PR #3214 完整报告

radixark/miles

docs: Harbor runs on Modal

合并时间: 2026-09-15 03:49

原文链接: <http://prhub.com.cn/radixark/miles/pull/3214>

PR#3214 分析报告: docs: Harbor runs on Modal

1. 执行摘要

这是一次严格意义上的单行文档改动: `docs/user-guide/sandbox-providers.md` 的 provider 能力对照表中, Modal 行的 Harbor 列由空单元格变成 $\checkmark$ (+1/-1), 用来宣告 Harbor 的 in-process 路径已在 Modal 上端到端跑通。代码零变更、回归风险为零, 但它在能力矩阵层面把 Modal 正式纳入 Harbor 支持列表, 而支撑这一格的证据只是一次手动运行, 且运行环境 (B200) 与 e2e 声明的 `hardware=["hopper"]` 并不一致, 这两点构成本 PR 真正值得关注的部分。

2. 功能与动机

动机在 PR body 里写得很直接: “Harbor's in-process path ran on Modal end to end, so the cell fills in.” 该表格的约定是——某格填 $\checkmark$ 表示“这个 provider $\times$ 运行路径组合上真实跑过一次 rollout”, 空单元格即代表未验证。作者于 2026-09-11 用 `test_harbor_rollout.py` 加 `HARBOR_ENV_TYPE=modal` 在 2xB200 上跑出 2 个 trial, 两者均正常提交、TITO strict gate 干净, 于是把这一格补齐。PR 同时声明 “No code change”, 并注明堆叠在 Harbor in-process 文档 PR 之上。

3. 实现拆解

- 变更入口: `docs/user-guide/sandbox-providers.md` 中 provider 比较表的 Modal 行、Harbor 列, 单元格由空改为 $\checkmark$; 同行其余三列、表头与列映射均未改动, reviewer 也确认 “the table structure and header/column mapping are unaffected”。
- 语义即承诺: 这不是排版微调, 而是把 Modal 从“理论可行”提升为“已实测”, 表格的 $\checkmark$ 会被下游使用者直接当作可用性结论。
- 证据链留在 PR 描述而非工程资产: 验证运行是 `tests/e2e/agent/test_harbor_rollout.py` + `HARBOR_ENV_TYPE=modal`, 2 个 trial 的 reward 都是 0.0 (agent 在 turn 上限内未完成任务, verifier 在沙箱内报断言失败, 单 trial 消耗 50k–115k 输入 token 的沙箱输出), 对照的 golden episode 是 reward 1.0、耗时 11 s。没有测试、CI job 或配置把这份结论固化下来。
- 披露但未落地的环境细节: B200 上 shipped `sglang_args` 在首次 prefill 即崩溃, 原因是 flashinfer 的 sm100 FMHA 没有 `head_dim=256` kernel, 本次运行额外加了 `--sglang-attention-backend triton`; 作者同时判定 e2e 的 `hardware=["hopper"]` 声明 “exactly right”, 因此未调整任何硬件声明, 也未把 workaround 写入代码或文档。

5. 测试 / 配置 / 部署配套：无。changed_files_count 为 1、commits 为 1，全部影响局限在该 Markdown 文件。

本次没有可执行代码变更，唯一“源码”是被修改的 Markdown 表格行。下面按现有表格语义重新整理了该行及其上下文说明，注释直接写在片段内。

<!-- docs/user-guide/sandbox-providers.md: provider x 运行路径的能力对照表。

表内自带约定——某格填 ✓ 代表“该组合上真的跑过一次 rollout”，

所以每一次勾选都是一次对外承诺，而不是理论可行性。 -->

| [Modal](https://modal.com/) | ✓ | | | ✓ |

<!-- ↑ Modal 行的 Harbor 列：本 PR 的唯一改动，由空单元格改为 ✓。

依据写在 PR body 而不是 CI 里：2026-09-11 在 2xB200 上以

HARBOR_ENV_TYPE=modal 跑 tests/e2e/agentice/test_harbor_rollout.py,

2 个 trial 均正常提交、TITO strict gate 干净；两个 trial 的 reward 都是 0.0

(agent 在 turn 上限内没完成任务，verifier 在沙箱内报断言失败)，

作为对照的 golden episode 是 reward 1.0、耗时 11 s。 -->

<!-- 同一行其余三列本次未动，表头与列映射保持不变。

需要留意的前提：该次运行额外加了 --sglang-attention-backend triton,

因为 B200 上 shipped sglang_args 会在首次 prefill 因

flashinfer sm100 FMHA 缺少 head_dim=256 kernel 而崩溃；

这条 workaround 没有写进表格，也没有写进代码。 -->

5. 评论区精华

claude[bot]: “LGTM — trivial documentation update.” 审查范围是 docs/user-guide/sandbox-providers.md 的单行 diff, “Confirmed the table structure and header/column mapping are unaffected, and the change is a pure markdown edit with no code or link changes.”

claude[bot] (同一段 review 的关键限界): “the PR description also provides supporting evidence (a rollout test with HARBOR_ENV_TYPE=modal) for the claim being documented, though that evidence itself isn't verifiable from the diff.”

claude[bot] (审查强度自陈): 这是 “exactly the kind of mechanical, self-contained edit (a single markdown table cell) that doesn't warrant deep review”, 安全风险 None, bug-hunting 系统无发现。

Shi-Dong: 直接 APPROVED, 无评论; 触发方式为 issue 评论 “@claude review”, 符合本仓库手动触发评审的配置。

6. 风险与影响

- 文档承诺强于证据：✓ 会被读作“Modal 可用”，但依据只有 2 个 trial 的单次手测，且两个 trial reward 均为 0.0——能证明链路连通，不能证明任务成功率；没有 CI 定期复跑，Modal 侧 SDK/ 镜像变化引发回归时无人察觉。
- 硬件口径不一致：B200 必须加 --sglang-attention-backend triton 才能跑通，而 tests/e2e/agentice/test_harbor_rollout.py 仍声明 hardware=["hopper"], 文档也未记录该

前置条件。Blackwell 用户在首次 prefill 就可能撞上文档未提及的崩溃，CI 覆盖的hopper 路径无法提前暴露。

- 可追溯性缺失：这一格的 ✓ 没有附带证据链接或运行记录，后续难以判断结论的时效与前提。
- 影响面：仅文档与选型决策；训练 /rollout 运行时行为完全不受影响，回归风险为零。

7. 关联脉络

- #3215：同一份 sandbox-providers.md 的表格，处理 Daytona 孤儿沙箱回收与存储配额 override——与本 PR 是能力矩阵上的并行演进（一个补 Daytona 运维语义，一个补 Modal 覆盖）。
- #3195：把沙箱测试从 e2b 专用改为凭据注册表驱动的任何 provider，本 PR 的 HARBOR_ENV_TYPE=modal 运行正建立在这一测试形态之上。
- #2837：Harbor in-process 模式的文档，本 PR 主张的正是该路径在 Modal 上可用，作者也说明本 PR 堆叠在其之上。
- #2876：Harbor 在真实 E2B 沙箱上的全链路 e2e，是“按 provider 逐个铺开验证”这条路线的前一站，Modal 是延续。
- #2701：为 CUDA 测试声明支持 GPU 世代的改造，hardware=["hopper"] 即来自它；本 PR 在 B200 上验证后仍认为 hopper 声明正确，凸显“实际验证硬件”与“声明硬件”之间的口径需要更清晰的约定。